PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-039752
(43)Date of publication of application : 13.02.1998

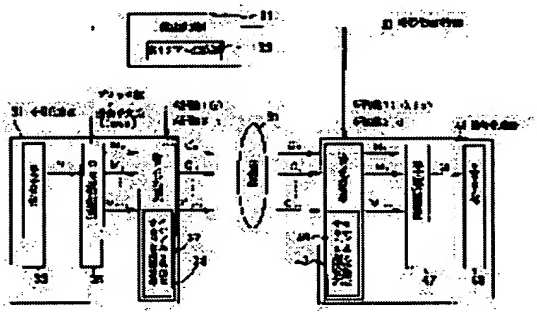
(51)Int.CI. G09C 1/00
G09C 1/00
G09C 1/00
H04L 9/08
H04L 9/30
H04L 9/32

(21)Application number : 08-189730
(22)Date of filing : 18.07.1996

(71)Applicant : NIPPON TELEGR & TELEPH CORP <NTT>
(72)Inventor : TAKAGI TAKESHI
NAITO SHOZO

(54) COMMUNICATION AND CERTIFICATION METHOD BY OPEN KEY CIPHER, AND DEVICE THEREFOR

(57)Abstract:
PROBLEM TO BE SOLVED: To provide a constitution method for an open key ciphering system and device therefor which has a strength of same level or more against a complete deciphering compared with a conventional open key cipher on a rational integer ring, and has a higher strength than ever against a broadcasting attack.
SOLUTION: A key forming device 21 forms prime ideals (p), (q) in an integer ring (O) on an algebraic number field for making them as a first secret key, and makes the remainders of their product (n)=(p) (q) as a first open key. Further, a second secret key d and a second open key e are formed from (p) and (q). A ciphering device 31 divides an inputted declarative sentence M into blocks, and ciphers them by performing a modulo ideal (n) raising operation to eth power, and outputs ciphered sentences (C0, C1,..., Cr-1) to a communication path 51. A decoding device 41 decodes the inputted blocks of the ciphered sentences by performing a modulo ideal (n) raising operation to dth power, and corporates the decoded blocks of the declarative sentence for outputting the declarative sentence.



LEGAL STATUS

[Date of request for examination] 26.12.2000
[Date of sending the examiner's decision of rejection] 08.04.2003
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]
[Patent number]
[Date of registration]
[Number of appeal against examiner's decision of rejection]
[Date of requesting appeal against examiner's decision of rejection]
[Date of extinction of right]

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号
特開平10－39752
(43)公開日　平成10年(1998)２月13日

(51)Int.Cl.⁶		識別記号	庁内整理番号	FI			技術表示箇所
G09C	1/00	630	7259－5J	G09C	1/00	630Z	
			7259－5J			630F	
		620	7259－5J			620B	
		640	7259－5J			640B	
H04L	9/08			H04L	9/00	601Z	

審査請求　未請求　請求項の数33　ＯＬ　（全 35 頁）　最終頁に続く

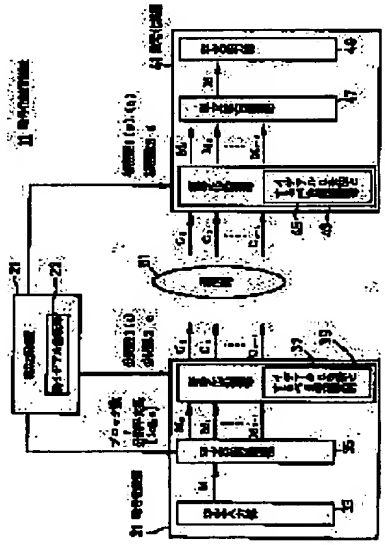
(21)出願番号　特願平8－189730

(22)出願日　平成8年(1996)７月18日

(71)出願人　000004226
日本電信電話株式会社
東京都新宿区西新宿三丁目19番２号
(72)発明者　高木　剛
東京都新宿区西新宿三丁目19番２号　日本電信電話株式会社内
(72)発明者　内藤　昭三
東京都新宿区西新宿三丁目19番２号　日本電信電話株式会社内
(74)代理人　弁理士　三好　秀和　（外１名）

(54)【発明の名称】　公開鍵暗号による通信および認証方法、ならびにそれらの装置

(57)【要約】
【課題】　従来の有理整数環上の公開鍵暗号と比較して、完全解読に対しては同程度以上の強度を持ち、同報通信攻撃に対して従来より高い強度を持つ公開鍵暗号方式および装置の構成法を提供する。
【解決手段】　鍵生成装置２１は、代数体上の整数環Ｏにおけるイデアル（ｐ）、（ｑ）を生成して第１の秘密鍵とし、その積（ｎ）＝（ｐ）（ｑ）の剰余類を第１の公開鍵とする。また（ｐ）、（ｑ）から第２の秘密鍵ｄと第２の公開鍵ｅを生成する。暗号化装置３１は、入力された平文Ｍをブロックに分割し、イデアル（ｎ）を法とするｅ乗演算により暗号化を行い、暗号文（Ｃ0，Ｃ1，…，Ｃr-1）を通信路５１に出力する。復号化装置４１は、入力された暗号文のブロックに対しイデアル（ｎ）を法とするｄ乗演算により復号化を行い、復号化された平文ブロックを統合して、平文Ｍを出力する。



【特許請求の範囲】

【請求項１】　ある素数データに基づいて、代数体上の整数環におけるイデアルであることを満たすような互いに異なる二つの自然数データの組である第１の秘密鍵データ、ならびに前記両自然数データの積である第１の公開鍵データを生成することを特徴とする公開鍵暗号の鍵生成方法。

【請求項２】　ある素数データに基づいて、代数体上の整数環におけるイデアルであることを満たすような互いに異なる二つの自然数データの組である第１の秘密鍵データ、ならびに前記両自然数データの積である第１の公開鍵データを生成し、

前記素数データのオイラー関数値の計算と、前記第１の秘密鍵データに関する最小公倍数演算と、を用いて、ともに自然数データである第２の秘密鍵データと第２の公開鍵データとを生成することを特徴とする公開鍵暗号の鍵生成方法。

【請求項３】　ある素数べきデータであるmc のオイラー関数値を求めて、

前記オイラー関数値をべき乗した結果に対してmc を法とする剰余を計算すると１となり、かつ、前記オイラー関数値より小さいいずれの自然数をべき乗した結果に対してmc を法とする剰余を計算しても、１にならない条件を満たすような、互いに異なる二つの自然数データであるpc とqc とを探索し、

pc とqc との組を第１の秘密鍵データとして出力し、

pc とqc を乗じてデータNc を求め、Nc を第１の公開鍵データとして出力することを特徴とする公開鍵暗号の鍵生成方法。

【請求項４】　前記pc とqc とにそれぞれ前記オイラー関数値をべき乗し１を減じた値同士の最小公倍数の値であるLc を算出し、

Lc を法とする剰余が１となるデータ値を積とするような二つの自然数データを探索し、

前記探索された二つの自然数データの一方を第２の秘密鍵データdc として、他方を第２の公開鍵データec として、それぞれ出力することを特徴とする請求項３に記載の公開鍵暗号の鍵生成方法。

【請求項５】　前記pc とqc との探索は、

ユークリッド的なmc 次円分体において、０でないイデアルによる、mc に対するオイラー関数の値を次数に持つ剰余類の完全代表系上の多次元ベクトルの原点からの距離である、ノルムを使用して行うことを特徴とする請求項３または４に記載の公開鍵暗号の鍵生成方法。

【請求項６】　前記pc とqc との探索は、

任意の円分体において、惰性する素数の積で表されるイデアルによる、mc に対するオイラー関数の値を次数に持つ剰余類の完全代表系上の多次元ベクトルの原点からの距離である、ノルムを使用して求めることによって行うことを特徴とする請求項３または４に記載の公開鍵暗号の鍵生成方法。

【請求項７】　二乗因子を含まないある有理整数データ値であるms に対して、４を法とする剰余を計算し、

この剰余の計算値が１となる場合は、判別式データの値をms とし、

この剰余の計算値が１以外の場合は、判別式データの値を４ms とし、

有理素数のなかから、この有理素数を法として前記判別式データを平方非剰余とするような互いに異なる二つの有理素数であるps とqs を探索し、

ps とqs の組を、第１の秘密鍵データとして出力し、

ps とqs を乗じてデータNs を求め、

Ns を第１の公開鍵データとして出力することを特徴とする公開鍵暗号の鍵生成方法。

【請求項８】　前記ps とqs のそれぞれの２乗から１を減じた値同士の最小公倍数データ値であるLs を算出し、

このLs を法とする剰余が１となるデータ値を積とするような二つの自然数データを探索し、

この二つの自然数データの一方を第２の秘密鍵データds とし、他方を第２の公開鍵データes として、それぞれ出力することを特徴とする請求項７に記載の公開鍵暗号の鍵生成方法。

【請求項９】　前記のps とqs の探索は、

ユークリッド的な２次体において、０でないイデアルによる２次元の剰余類の完全代表系である２次元ベクトルの原点からの距離である、ノルムを使用して求めることによって行うことを特徴とする請求項７または請求項８に記載の公開鍵暗号の鍵生成方法。

【請求項１０】　ある素数べきデータであるmc を入力する入力手段と、

mc のオイラー関数値を求めるオイラー関数計算手段と、

前記オイラー関数値をべき乗した結果に対してmc を法とする剰余を計算すると１となり、かつ、前記オイラー関数値より小さいいずれの自然数をべき乗した結果に対してmc を法とする剰余を計算しても、１にならない条件を満たすような、互いに異なる二つの自然数データであるpc とqc とを探索する第１の探索手段と、

pc とqc を乗じてデータNc を求める乗算手段と、

pc とqc とにそれぞれ前記オイラー関数値をべき乗し１を減じた値同士の最小公倍数データ値であるLc を算出し、Lc を法とする剰余が１となるデータ値を積とするような二つの自然数データdc およびec を探索する第２の探索手段と、

pc とqc との組を第１の秘密鍵データとして、Nc を第１の公開鍵データとして、dc を第２の秘密鍵データとして、ec を第２の公開鍵データとして、それぞれ出力する出力手段と、

を具備したことを特徴とする公開鍵暗号の鍵生成装置。

【請求項１１】　二乗因子を含まないある有理整数データ値であるｍｓ を入力する入力手段と、
ｍｓ に対して、４を法とする剰余を計算し、この剰余の計算値が１となる場合は、判別式データの値をｍｓ とし、この剰余の計算値が１以外の場合は、判別式データの値を４ｍｓ とする判別式データ設定手段と、
有理素数のなかから、この有理素数を法として前記判別式データを平方非剰余とするような互いに異なる二つの有理素数であるｐｓ とｑｓ とを探索する第１の探索手段と、
ｐｓ とｑｓ を乗じてデータＮｓ を求める乗算手段と、
ｐｓ とｑｓ のそれぞれの２乗から１を退じた値同士の最小公倍数データ値であるＬｓ を算出し、このＬｓ を法とする剰余が１となるデータ値を核とするような二つの自然数データｄｓ およびｅｓ を探索する第２の探索手段と、
ｐｓ とｑｓ の組を第１の秘密鍵データとして、Ｎｓ を第１の公開鍵データとして、ｄｓ を第２の秘密鍵データとして、ｅｓ を第２の公開鍵データとして、それぞれ出力する出力手段と、
を具備したことを特徴とする公開鍵暗号の鍵生成装置。
【請求項１２】　第１および第２の公開鍵により平文データを暗号化して暗号文データを生成する公開鍵暗号による暗号化方法であって、
平文データを複数ブロックに分割して得られた平文ブロックデータに対して、第２の公開鍵データ値であるｅｃ またはｅｓ をべき乗した値を計算し、
この計算値に対して、第１の公開鍵であるイデアルＮｃ またはＮｓ を法とした剰余を求めることによって、暗号文データを生成することを特徴とする公開鍵暗号による暗号化方法。
【請求項１３】　前記イデアルを法とする剰余類の取り方は、
ユークリッド的なｍｃ 次円分体において、０でないイデアルＮｃ を法とする剰余類の完全代表系として、ｍｃ に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とし、
前記平文データが、それぞれ前記ｍｃ に対するオイラー関数値に等しいブロック数からなる平文ブロックデータに分割された後に暗号化されることを特徴とする請求項１２に記載の公開鍵暗号による暗号化方法。
【請求項１４】　前記イデアルを法とする剰余類の取り方は、
ユークリッド的な２次体において、０でないイデアルＮｓ を法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とし、
前記平文データが、それぞれ２つのブロックからなる平文ブロックデータに分割された後に暗号化されることを特徴とする請求項１２に記載の公開鍵暗号による暗号化方法。
【請求項１５】　平文ブロックデータに対して、第２の公開鍵データ値であるｅｃ をべき乗した値を計算するべき乗計算手段と、
この計算値に対して、第１の公開鍵であるイデアルＮｃ を法とした剰余を求める剰余類計算手段と、
を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的なｍｃ 次円分体において、０でないイデアルＮｃ を法とする剰余類の完全代表系として、ｍｃ に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを特徴とする公開鍵暗号による暗号化装置。
【請求項１６】　平文ブロックデータに対して、第２の公開鍵データ値であるｅｓ をべき乗した値を計算するべき乗計算手段と、
この計算値に対して、第１の公開鍵であるイデアルＮｓ を法とした剰余を求める剰余類計算手段と、
を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な２次体において、０でないイデアルＮｓ を法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを特徴とする公開鍵暗号による暗号化装置。
【請求項１７】　請求項１２ないし請求項１４のいずれか１項記載の暗号化方法により生成された暗号文ブロックデータに対して復号化を施して平文データを生成する公開鍵暗号の復号化方法であって、
前記暗号文ブロックデータに対して、第２の秘密鍵データ値であるｄｃ またはｄｓ をべき乗した値を計算し、
この計算値に対して、第１の公開鍵であるイデアルＮｃ またはＮｓ を法とした剰余を求めることによって、平文データを生成することを特徴とする公開鍵暗号の復号化方法。
【請求項１８】　前記イデアルを法とする剰余類の取り方は、
ユークリッド的なｍｃ 次円分体において、０でないイデアルＮｃ を法とする剰余類の完全代表系として、ｍｃ に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを特徴とする請求項１７に記載の公開鍵暗号の復号化方法。
【請求項１９】　前記イデアルを法とする剰余類の取り方は、
ユークリッド的な２次体において、０でないイデアルＮｓ を法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを特徴とする請求項１７に記載の公開鍵暗号の復号化方法。
【請求項２０】　暗号文ブロックデータに対して、第２

の秘密鍵データ値であるd_cをべき乗した値を計算するべき乗計算手段と、

この計算値に対して、第１の公開鍵であるイデアルN_cを法とした剰余を求める剰余類計算手段と、

を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的なm_c次円分体において、０でないイデアルN_cを法とする剰余類の完全代表系として、m_cに対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを特徴とする公開鍵暗号の復号化装置。

【請求項２１】　暗号文ブロックデータに対して、第２の秘密鍵データ値であるd_sをべき乗じた値を計算するべき乗計算手段と、

この計算値に対して、第１の公開鍵であるイデアルN_sを法とした剰余を求める剰余類計算手段と、

を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な２次体において、０でないイデアルN_sを法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを特徴とする公開鍵暗号の復号化装置。

【請求項２２】　第１の公開鍵および第２の秘密鍵により、認証文から暗号化された認証子データを生成する公開鍵暗号方式による認証文生成方法であって、

前記認証文をハッシュ化し、複数ブロックに分割した認証子ブロックデータを生成し、

この認証子ブロックデータに対して、第２の秘密鍵データ値であるd_cまたはd_sをべき乗した値を計算し、

この計算値に対して、第１の公開鍵データであるイデアルN_cまたはN_sを法とした剰余を求めることによって、暗号化認証子データを生成することを特徴とする公開鍵暗号方式による認証文生成方法。

【請求項２３】　前記イデアルを法とする剰余類の取り方は、

ユークリッド的なm_c次円分体において、０でないイデアルN_cを法とする剰余類の完全代表系として、m_cに対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とし、

前記平文データが、それぞれ前記m_cに対するオイラー関数値に等しいブロック数からなる平文ブロックデータに分割された後に暗号化されることを特徴とする請求項２２に記載の公開鍵暗号方式による認証文生成方法。

【請求項２４】　前記イデアルを法とする剰余類の取り方は、

ユークリッド的な２次体において、０でないイデアルN_sを法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とし、

前記平文データが、それぞれ２つのブロックからなる平文ブロックデータに分割された後に暗号化されることを特徴とする請求項２２に記載の公開鍵暗号方式による認証文生成方法。

【請求項２５】　入力された認証文をハッシュ化した後に分割し、それぞれm_cに対するオイラー関数値に等しいブロック数からなる認証子ブロックデータを生成する認証子データ生成手段と、

この生成された認証子ブロックデータに対して、第２の秘密鍵データ値であるd_cをべき乗した値を計算するべき乗計算手段と、

この計算値に対して、第１の公開鍵データであるイデアルN_cを法とした剰余を求める剰余類計算手段と、

を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的なm_c次円分体において、０でないイデアルN_cを法とする剰余類の完全代表系として、m_cに対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを特徴とする公開鍵暗号方式による認証文生成装置。

【請求項２６】　入力された認証文をハッシュ化した後に分割し、それぞれ２つのブロックからなる認証子ブロックデータを生成する認証子データ生成手段と、

この生成された認証子ブロックデータに対して、第２の秘密鍵データ値であるd_sをべき乗した値を計算するべき乗計算手段と、

この計算値に対して、第１の公開鍵データであるイデアルN_sを法とした剰余を求めて暗号化認証子データを得る剰余類計算手段と、

を備えてなり、前記剰余類計算手段におけるイデアルN_sを法とする剰余類の取り方は、ユークリッド的な２次体において、０でないイデアルN_sを法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを特徴とする公開鍵暗号方式による認証文生成装置。

【請求項２７】　請求項２２ないし請求項２４のいずれか１項記載の認証文生成方法により生成された暗号化認証子データを復号化し、平文でありこの暗号化認証子データに相当する認証文データの正当性を検証する認証文検証方法であって、

暗号化された認証子ブロックデータに対して、第２の公開鍵データ値であるe_cまたはe_sをべき乗した値を計算し、

この計算値に対して、第１の公開鍵であるイデアルN_cまたはN_sを法とした剰余を求めることによって、認証子データを復号化し、

この復号化した認証子データと、前記平文の認証子データとが一致していた場合は、認証ないし検証過程を正当と判定し、

この復号化した認証子データと、前記平文の認証子データとが不一致であった場合は、認証ないし検証過程を正

当でないと判定する
ことを特徴とする公開鍵暗号方式による認証文検証方法。
【請求項２８】　前記イデアルを法とする剰余類の取り方は、
ユークリッド的なｍｃ次円分体において、０でないイデアルＮｃを法とする剰余類の完全代表系として、ｍｃに対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを特徴とする請求項２７に記載の公開鍵暗号方式による認証文検証方法。
【請求項２９】　前記イデアルを法とする剰余類の取り方は、
ユークリッド的な２次体において、０でないイデアルＮｓを法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを特徴とする請求項２７に記載の公開鍵暗号方式による認証文検証方法。
【請求項３０】　暗号化された認証子ブロックデータに対して、第２の公開鍵データ値であるｅｃをべき乗した値を計算するべき乗計算手段と、
この計算値に対して、第１の公開鍵であるイデアルＮｃを法とした剰余を求めて復号化された認証子データを得る剰余類計算手段と、
この復号化された認証子データおよびこれに対応する平文の認証子データの比較に基づいて認証判定を行う判定手段と、
を備える公開鍵暗号方式による認証文検証装置であって、
前記剰余類計算手段における前記イデアルを法とする剰余類の取り方は、
ユークリッド的なｍｃ次円分体において、０でないイデアルＮｃを法とする剰余類の完全代表系として、ｍｃに対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを特徴とする公開鍵暗号方式による認証文検証装置。
【請求項３１】　暗号化された認証子ブロックデータに対して、第２の公開鍵データ値であるｅｓをべき乗した値を計算するべき乗計算手段と、
この計算値に対して、第１の公開鍵であるイデアルＮｓを法とした剰余を求めて復号化された認証子データを得る剰余類計算手段と、
この復号化された認証子データおよびこれに対応する平文の認証子データの比較に基づいて認証判定を行う判定手段と、
を備える公開鍵暗号方式による認証文検証装置であって、
前記剰余類計算手段における前記イデアルを法とする剰余類の取り方は、
ユークリッド的な２次体において、０でないイデアルＮｓを法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを特徴とする公開鍵暗号方式による認証文検証装置。
【請求項３２】　前記剰余類計算手段における剰余の計算は、前記ｍｃに対するオイラー関数値に等しい数の各成分毎に行うことを特徴とする請求項１５、請求項２０、請求項２５、または請求項３０のいずれか１項に記載の装置。
【請求項３３】　前記剰余類計算手段における剰余の計算は、２つの各成分毎に行うことを特徴とする請求項１６、請求項２１、請求項２６、または請求項３１のいずれか１項に記載の装置。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明は、暗号化鍵が公開され、復号化鍵のみが秘匿される公開鍵方式の暗号に係り、特に、従来のリベスト・シャミール・アドルマン暗号（以下、ＲＳＡ暗号と略す）より同報通信攻撃に対する強度を高めた公開鍵暗号による通信および認証方式並びにそれらの装置に関する。
【０００２】
【従来の技術】電気通信において、通信内容の盗聴を防き、改竄を検出するために暗号技術は不可欠である。特に、鍵管理が簡便な公開鍵暗号が有効であり、広く利用されつつある。代表的な公開鍵暗号のアルゴリズムとして、べき（冪）乗剰余演算を用いるＲＳＡ暗号があり、既に実用化されている。
【０００３】以下、池野、小山による「現代暗号理論」（電子情報通信学会発行）に基づいて、ＲＳＡ暗号の基本原理、その鍵生成法および認証方法について説明する。
（ＲＳＡ暗号の基本原理）暗号化鍵は（ｅ，Ｎ）の組であり、対応する復号化鍵は（ｄ，Ｎ）の組である。
【０００４】ｅとＮとは公開鍵であり、ｄは秘密鍵である。平文をＭ、暗号文をＣとする。暗号化Ｅと復号化Ｄのアルゴリズムは、次の式（１）および式（２）で表される。
【数１】
$$C = E(M) = M^e \bmod N \quad \cdots (1)$$
$$M = D(C) = C^d \bmod N \quad \cdots (2)$$
但し、ＭとＣとは０からＮ－１の間の整数である。もし元のメッセージがＮより大きければ、サイズＮのブロックに分割して逐一、暗号化、復号化を適用すればよい。
【０００５】暗号化と復号化は、１対１かつ上への写像である。従って、ＭとＣを代表してＭで表すと、
【数２】
$$D(E(M)) = E(D(M)) = M \quad \cdots (3)$$
式（３）が成立する。具体的には、
【数３】

$$Med \equiv M \bmod N \quad \cdots (4)$$

式（4）が成立する。但し式（4）および以下の記述において、「≡」は合同を示す記号とする。式（4）がすべてのMに対して成立するような、暗号鍵e，d，Nの生成手順は以下のとおりである。

【0006】〔RSA暗号の鍵生成〕まず、任意の相異なる二つの大きな素数p，qを選び、その積N＝p・qを計算する（第1段階）。

【0007】次いで、（p－1）と（q－1）の最小公倍数Lを計算し、Lと互いに素でより小さな任意の整数eを選ぶ（第2段階）。

【0008】

【数4】

$$L = LCM((p-1),(q-1)) \quad \cdots (5)$$

$$GCD(e, L) = 1, \quad 1 < e < L \quad \cdots (6)$$

次いで、第2段階で求めたeとLをもとに、次の合同式（7）を解き、dを求める（第3段階）。

【0009】

【数5】

$$e \cdot d \equiv 1 \bmod L \quad \cdots (7)$$

式（7）からdを求めるには、ユークリッドの互除法を用いればよい。また、前記の生成手順は、まず、第2段階でeを先に選び、第3段階でdを計算しているが、逆にdを先に選び、後でeを計算してもよい。

【0010】このような、従来のRSA型公開鍵による暗号化通信装置の構成例を図33に示す。

【0011】〔RSA暗号による認証〕RSA暗号による認証通信は、以下のようにして行われる。まず、送信者は、認証文Mをハッシュ関数hによりハッシュ化し、認証子h（M）を得る。次いで認証子h（M）を自己の秘密鍵dで暗号化し、暗号化認証子h（C）を得る。次いで、暗号化認証子h（C）と認証文Mの組を送信者から受信者に送る。

【0012】

【数6】

$$h(C) \equiv (h(M))^d \bmod N \quad \cdots (8)$$

受信者は、暗号化認証子h（C）と認証文Mの組を受信すると、送信者の公開鍵eを使用して、暗号化認証子h（C）を復号化し認証子h（M）を得る。

【0013】

【数7】

$$h(M) \equiv (h(C))^e \bmod N \quad \cdots (9)$$

次いで、受信した認証文Mをハッシュ関数hによりハッシュ化し、認証子h（M）＊を得る。そして復号化した認証子h（M）とハッシュ化した認証子h（M）＊とを比較して正当性を判断する。すなわち、両者が一致していれば認証文が正当であり、不一致であれば不当と判断する。

【0014】このように、RSA型の公開鍵暗号装置は、公開鍵と秘密鍵の演算を逆に適用することにより、認証通信装置としても使用することができる。

【0015】なお、この認証通信において、前記暗号化認証子h（C）と認証文Mの組をさらに受信者の公開鍵を用いて暗号化する暗号化認証通信も、認証通信と暗号化通信とを組み合わせることによって可能であることはいうまでもない。

【0016】ところで、一般に暗号技術の性能評価の尺度は、暗号を解読しようとする攻撃に対する安全性の強度と、暗号化・復号化の速度である。強度が高く、速度の速い暗号が優秀な暗号である。

【0017】RSA暗号などの公開鍵暗号は、暗号化鍵である公開された公開鍵から、復号化鍵である秘密鍵を得ることが計算量的に困難であることに安全性の根拠を置いている。

【0018】RSA暗号では、公開鍵が素因数分解できれば、暗号文から平文を得ることができることが、文献「R. L. Rivest, A. Shamir and L. Adleman; "A method for obtaining digital signatures and public-key cryptosystems," Comm. ACM, vol. 21, No. 2, pp. 120-126, (1978)」に示されている。これらの安全性の評価は、完全解読を対象としたものであり、この場合には公開鍵の素因数分解の十分性あるいは計算量的な同値性が示されている。

【0019】

【発明が解決しようとする課題】しかしながら、J. Hastadは、文献「J. Hastad; "On using RSA with low exponent in a public key network", Proceeding of CRYPTO '85, Springer-Verlag (1988)」において、同報通信攻撃と呼ばれるRSAタイプの公開鍵暗号に対する解読法を提案した。

【0020】これにより、RSAタイプの公開鍵暗号では、同一の発信者から複数の受信者に対して、同一内容の平文をそれぞれ異なる公開鍵により暗号化して送信された同報通信が盗聴されると、複数の暗号文から公開鍵を素因数分解することなく平文を得ることができることがわかり、安全性の評価基準を見直す必要がでてきた。

【0021】RSA暗号では暗号化のべき指数を大きくすることにより同報通信攻撃を回避できるが、べき指数を大きくするとべき乗計算時間および剰余計算時間が増加し、暗号化の速度が遅くなるという問題点があった。

【0022】このオリジナルなRSA暗号の同報通信攻撃に対する防御を強化した暗号として、楕円曲線を用いたRSA暗号が知られている。この楕円曲線を用いたRSA暗号は、文献「森門 秀典，小山 謙二，"3次曲線上のRSA型暗号方式の同報通信における安全性"，信学技報ISEC94-10，(1994)」において、その同報通信攻撃に対する安全強度が評価されているが、暗号化および復号化の速度がオリジナルのRSA暗号と比較して5倍以上遅くなり、用途が限定されるという問題点があった。

【００２３】以上の問題点に鑑み、本発明の第１の課題は、従来の有理整数環上のＲＳＡ暗号と比較して同程度以上の強度を持ち、同報通信攻撃に対して従来以上の強い暗号方式および装置の構成法を与えることである。

【００２４】また、本発明の第２の課題は、楕円曲線上のＲＳＡ暗号と比較して、暗号化および復号化処理が高速である暗号方式および装置の構成法を与えることである。

【００２５】また、本発明の第３の課題は、認証装置としても利用可能であり、ひとつの装置で暗号通信と認証通信の両用が可能となるような暗号および認証方式および装置の構成法を与えることである。

【００２６】

【課題を解決するための手段】上記課題を解決するため、本発明においては、従来の有理整数環上でのＲＳＡ暗号を代数体の整数環の上へ拡張した公開鍵暗号方式の具体的な構成方法である、代数体の整数環上のイデアルを用いた鍵生成方法、べき乗演算方法、およびイデアルを法とする剰余類の演算方法を与え、さらに鍵生成装置、暗号通信装置および認証通信装置として実現する。

【００２７】すなわち、請求項１記載の発明は、ある素数データに基づいて、代数体上の整数環における素イデアルであることを満たすような互いに異なる二つの自然数データの組である第１の秘密鍵データ、ならびに前記両自然数データの積である第１の公開鍵データを生成することを要旨とする公開鍵暗号の鍵生成方法である。

【００２８】また、請求項２記載の発明は、ある素数データに基づいて、代数体上の整数環における素イデアルであることを満たすような互いに異なる二つの自然数データの組である第１の秘密鍵データ、ならびに前記両自然数データの積である第１の公開鍵データを生成し、前記素数データのオイラー関数値の計算と、前記第１の秘密鍵データに関する最小公倍数演算と、を用いて、ともに自然数データである第２の秘密鍵データと第２の公開鍵データとを生成することを要旨とする公開鍵暗号の鍵生成方法である。

【００２９】また、請求項３記載の発明は、ある素数べきデータであるmcのオイラー関数値を求めて、前記オイラー関数値をべき乗した結果に対してmcを法とする剰余を計算すると１となり、かつ、前記オイラー関数値より小さいいずれの自然数をべき乗した結果に対してmcを法とする剰余を計算しても、１にならない条件を満たすような、互いに異なる二つの自然数データであるpcとqcとを探索し、pcとqcとの組を第１の秘密鍵データとして出力し、pcとqcを乗じてデータNcを求め、Ncを第１の公開鍵データとして出力することを要旨とする公開鍵暗号の鍵生成方法である。

【００３０】また、請求項４記載の発明は、請求項３に記載の公開鍵暗号の鍵生成方法において、前記pcとqcとにそれぞれ前記オイラー関数値をべき乗し１を減じた値同士の最小公倍数の値であるLcを算出し、Lcを法とする剰余が１となるデータ値を核とするような二つの自然数データを探索し、前記探索された二つの自然数データの一方を第２の秘密鍵データdcとして、他方を第２の公開鍵データecとして、それぞれ出力することを要旨とする。

【００３１】また、請求項５記載の発明は、請求項３または４に記載の公開鍵暗号の鍵生成方法において、前記pcとqcとの探索は、ユークリッド的なmc次円分体において、０でないイデアルによる、mcに対するオイラー関数の値を次数に持つ剰余類の完全代表系上の多次元ベクトルの原点からの距離である、ノルムを使用して行うことを要旨とする。

【００３２】また、請求項６記載の発明は、請求項３または４に記載の公開鍵暗号の鍵生成方法において、前記pcとqcとの探索は、任意の円分体において、惟性する素数の積であらされるイデアルによる、mcに対するオイラー関数の値を次数に持つ剰余類の完全代表系上の多次元ベクトルの原点からの距離である、ノルムを使用して求めることによって行うことを要旨とする。

【００３３】また、請求項７記載の発明は、二乗因子を含まないある有理整数データ値であるmsに対して、４を法とする剰余を計算し、この剰余の計算値が１となる場合は、判別式データの値をmsとし、この剰余の計算値が１以外の場合は、判別式データの値を４msとし、有理整数のなかから、この有理整数を法として前記判別式データを平方非剰余とするような互いに異なる二つの数であるpsとqsを探索し、psとqsの組を第１の秘密鍵データとして出力し、psとqsを乗じてデータNsを求め、Nsを第１の公開鍵データとして出力することを要旨とする公開鍵暗号の鍵生成方法である。

【００３４】また、請求項８記載の発明は、請求項７に記載の公開鍵暗号の鍵生成方法において、前記psとqsのそれぞれの２乗から１を減じた値同士の最小公倍数データ値であるLsを算出し、このLsを法とする剰余が１となるデータ値を核とするような二つの自然数データを探索し、この二つの自然数データの一方を第２の秘密鍵データdsとし、他方を第２の公開鍵データesとして、それぞれ出力することを要旨とする。

【００３５】また、請求項９記載の発明は、請求項７または請求項８に記載の公開鍵暗号の鍵生成方法において、前記のpsとqsの探索は、ユークリッド的な２次体において、０でないイデアルによる２次の剰余類の完全代表系である２次元ベクトルの原点からの距離である、ノルムを使用して求めることによって行うことを要旨とする。

【００３６】また、請求項１０記載の発明は、ある素数べきデータであるmcを入力する入力手段と、mcのオイラー関数値を求めるオイラー関数計算手段と、前記オイラー関数値をべき乗した結果に対してmcを法とする

剰余を計算すると１となり、かつ、前記オイラー関数値より小さいいずれの自然数をべき乗した結果に対してｍｏを法とする剰余を計算しても、１にならない条件を満たすような、互いに異なる二つの自然数データであるｐｏとｑｏとを探索する第１の探索手段と、ｐｏとｑｏを乗じてデータＮｏを求める乗算手段と、ｐｏとｑｏとにそれぞれ前記オイラー関数値をべき乗し１を減じた値同士の最小公倍数データ値であるＬｏを算出し、Ｌｏを法とする剰余が１となるデータ値を核とするような二つの自然数データｄｏおよびｅｏを探索する第２の探索手段と、ｐｏとｑｏとの組を第１の秘密鍵データとして、Ｎｏを第１の公開鍵データとして、ｄｏを第２の秘密鍵データとして、ｅｏを第２の公開鍵データとして、それぞれ出力する出力手段と、を具備したことを要旨とする公開鍵暗号の鍵生成装置である。

【００３７】また、請求項１１記載の発明は、二乗因子を含まないある有理整数データ値であるｍｓを入力する入力手段と、ｍｓに対して、４を法とする剰余を計算し、この剰余の計算値が１となる場合は、判別式データの値をｍｓとし、この剰余の計算値が１以外の場合は、判別式データの値を４ｍｓとする判別式データ設定手段と、有理素数のなかから、この有理素数を法として前記判別式データを平方非剰余とするような互いに異なる二つの有理素数であるｐｓとｑｓとを探索する第１の探索手段と、ｐｓとｑｓを乗じてデータＮｓを求める乗算手段と、ｐｓとｑｓのそれぞれの２乗から１を減じた値同士の最小公倍数データ値であるＬｓを算出し、このＬｓを法とする剰余が１となるデータ値を核とするような二つの自然数データｄｓおよびｅｓを探索する第２の探索手段と、ｐｓとｑｓの組を第１の秘密鍵データとして、Ｎｓを第１の公開鍵データとして、ｄｓを第２の秘密鍵データとして、ｅｓを第２の公開鍵データとして、それぞれ出力する出力手段と、を具備したことを要旨とする公開鍵暗号の鍵生成装置。

【００３８】また、請求項１２記載の発明は、第１および第２の公開鍵により平文データを暗号化して暗号文データを生成する公開鍵暗号による暗号化方法であって、平文データを複数ブロックに分割して得られた平文ブロックデータに対して、第２の公開鍵データ値であるｅｏまたはｅｓをべき乗した値を計算し、この計算値に対して、第１の公開鍵であるイデアルＮｏまたはＮｓを法とした剰余を求めることによって、暗号文データを生成することを要旨とする公開鍵暗号による暗号化方法である。

【００３９】また、請求項１３記載の発明は、請求項１２に記載の公開鍵暗号による暗号化方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的なｍｏ次円分体において、０でないイデアルＮｏを法とする剰余類の完全代表系として、ｍｏに対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とし、前記平文データが、それぞれ前記ｍｏに対するオイラー関数値に等しいブロック数からなる平文ブロックデータに分割された後に暗号化されることを要旨とする。

【００４０】また、請求項１４記載の発明は、請求項１２に記載の公開鍵暗号による暗号化方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的な２次体において、０でないイデアルＮｓを法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とし、前記平文データが、それぞれ２つのブロックからなる平文ブロックデータに分割された後に暗号化されることを要旨とする。

【００４１】また、請求項１５記載の発明は、平文ブロックデータに対して、第２の公開鍵データ値であるｅｏをべき乗した値を計算するべき乗計算手段と、この計算値に対して、第１の公開鍵であるイデアルＮｏを法とした剰余を求める剰余類計算手段と、を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的なｍｏ次円分体において、０でないイデアルＮｏを法とする剰余類の完全代表系として、ｍｏに対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする公開鍵暗号による暗号化装置である。

【００４２】また、請求項１６記載の発明は、平文ブロックデータに対して、第２の公開鍵データ値であるｅｓをべき乗した値を計算するべき乗計算手段と、この計算値に対して、第１の公開鍵であるイデアルＮｓを法とした剰余を求める剰余類計算手段と、を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な２次体において、０でないイデアルＮｓを法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを要旨とする公開鍵暗号による暗号化装置である。

【００４３】また、請求項１７記載の発明は、請求項１２ないし請求項１４のいずれか１項記載の暗号化方法により生成された暗号文ブロックデータに対して復号化を施して平文データを生成する公開鍵暗号の復号化方法であって、前記暗号文ブロックデータに対して、第２の秘密鍵データ値であるｄｏまたはｄｓをべき乗した値を計算し、この計算値に対して、第１の公開鍵であるイデアルＮｏまたはＮｓを法とした剰余を求めることによって、平文データを生成することを要旨とする公開鍵暗号の復号化方法である。

【００４４】また、請求項１８記載の発明は、請求項１７に記載の公開鍵暗号の復号化方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的なｍｏ次円分体において、０でないイデアルＮｏを法とする剰余類の完全代表系として、ｍｏに対するオイラー関数

の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする。

【００４５】また、請求項１９記載の発明は、請求項１７に記載の公開鍵暗号の復号化方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的な２次体において、０でないイデアルNs を法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを要旨とする。

【００４６】また、請求項２０記載の発明は、暗号文ブロックデータに対して、第２の秘密鍵データ値であるｄｃ をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第１の公開鍵であるイデアルＮｃ を法とした剰余を求める剰余類計算手段と、を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的なmc 次円分体において、０でないイデアルＮｃ を法とする剰余類の完全代表系として、mc に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする公開鍵暗号の復号化装置である。

【００４７】また、請求項２１記載の発明は、暗号文ブロックデータに対して、第２の秘密鍵データ値であるｄｓ をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第１の公開鍵であるイデアルＮｓ を法とした剰余を求める剰余類計算手段と、を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な２次体において、０でないイデアルＮｓ を法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを要旨とする公開鍵暗号の復号化装置である。

【００４８】また、請求項２２記載の発明は、第１の公開鍵および第２の秘密鍵により、認証文から暗号化された認証子データを生成する公開鍵暗号方式による認証文生成方法であって、前記認証文をハッシュ化し、複数ブロックに分割した認証子ブロックデータを生成し、この認証子ブロックデータに対して、第２の秘密鍵データ値であるｄｃ またはｄｓ をべき乗した値を計算し、この計算値に対して、第１の公開鍵データであるイデアルＮｃ またはＮｓ を法とした剰余を求めることによって、暗号化認証子データを生成することを要旨とする公開鍵暗号方式による認証文生成方法である。

【００４９】また、請求項２３記載の発明は、請求項２２に記載の公開鍵暗号による認証文生成方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的なmc 次円分体において、０でないイデアルＮｃ を法とする剰余類の完全代表系として、mc に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とし、前記平文データが、それぞれ前記mc に対するオイラー関数値に等しいブロック数からなる平文ブロックデータに分割された後に暗号化されることを要旨とする。

【００５０】また、請求項２４記載の発明は、請求項２２に記載の公開鍵暗号による認証文生成方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的な２次体において、０でないイデアルＮｓ を法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とし、前記平文データが、それぞれ２つのブロックからなる平文ブロックデータに分割された後に暗号化されることを要旨とする。

【００５１】また、請求項２５記載の発明は、入力された認証文をハッシュ化した後に分割し、それぞれmc に対するオイラー関数値に等しいブロック数からなる認証子ブロックデータを生成する認証子データ生成手段と、この生成された認証子ブロックデータに対して、第２の秘密鍵データ値であるｄｃ をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第１の公開鍵データであるイデアルＮｃ を法とした剰余を求める剰余類計算手段と、を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的なmc 次円分体において、０でないイデアルＮｃ を法とする剰余類の完全代表系として、mcに対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする公開鍵暗号による認証文生成装置である。

【００５２】また、請求項２６記載の発明は、入力された認証文をハッシュ化した後に分割し、それぞれ２つのブロックからなる認証子ブロックデータを生成する認証子データ生成手段と、この生成された認証子ブロックデータに対して、第２の秘密鍵データ値であるｄｓ をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第１の公開鍵データであるイデアルＮｓ を法とした剰余を求めて暗号化認証子データを得る剰余類計算手段と、を備えてなり、前記剰余類計算手段におけるイデアルＮｓ を法とする剰余類の取り方は、ユークリッド的な２次体において、０でないイデアルＮｓ を法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを要旨とする公開鍵暗号による認証文生成装置である。

【００５３】また、請求項２７記載の発明は、請求項２２ないし請求項２４のいずれか１項記載の認証文生成方法により生成された暗号化認証子データを復号化し、平文でありこの暗号化認証子データに相当する認証文データの正当性を検証する認証文検証方法であって、暗号化された認証子ブロックデータに対して、第２の公開鍵データ値であるｅｃ またはｅｓ をべき乗した値を計算し、この計算値に対して、第１の公開鍵であるイデアルＮｃ またはＮｓ を法とした剰余を求めることによって、認証

子データを復号化し、この復号化した認証子データと、前記平文の認証子データとが一致していた場合は、認証ないし検証過程を正当と判定し、この復号化した認証子データと、前記平文の認証子データとが不一致であった場合は、認証ないし検証過程を正当でないと判定する、ことを要旨とする公開鍵暗号による認証文検証方法である。

【００５４】また、請求項２８記載の発明は、請求項２７に記載の公開鍵暗号による認証文検証方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的なmc 次円分体において、０でないイデアルNc を法とする剰余類の完全代表系として、mc に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする。

【００５５】また、請求項２９記載の発明は、請求項２７に記載の公開鍵暗号による認証文検証方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的な２次体において、０でないイデアルNs を法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを要旨とする。

【００５６】また、請求項３０記載の発明は、暗号化された認証子ブロックデータに対して、第２の公開鍵データ値であるec をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第１の公開鍵であるイデアルNc を法とした剰余を求めて復号化された認証子データを得る剰余類計算手段と、この復号化された認証子データおよびこれに対応する平文の認証子データの比較に基づいて認証判定を行う判定手段と、を備える認証文検証装置であって、前記剰余類計算手段における前記イデアルを法とする剰余類の取り方は、ユークリッド的なmc 次円分体において、０でないイデアルNc を法とする剰余類の完全代表系として、mc に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする公開鍵暗号による認証文検証装置である。

【００５７】また、請求項３１記載の発明は、暗号化された認証子ブロックデータに対して、第２の公開鍵データ値であるes をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第１の公開鍵であるイデアルNs を法とした剰余を求めて復号化された認証子データを得る剰余類計算手段と、この復号化された認証子データおよびこれに対応する平文の認証子データの比較に基づいて認証判定を行う判定手段と、を備える認証文検証装置であって、前記剰余類計算手段における前記イデアルを法とする剰余類の取り方は、ユークリッド的な２次体において、０でないイデアルNs を法とする剰余類の完全代表系として、２次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを要旨とする公開鍵暗号による認証文検証装置である。

【００５８】また、請求項３２記載の発明は、請求項１５、請求項２０、請求項２５、または請求項３０のいずれか１項に記載の装置において、前記剰余類計算手段における剰余の計算は、前記mc に対するオイラー関数値に等しい数の各成分毎に行うことを要旨とする。

【００５９】また、請求項３３記載の発明は、請求項１６、請求項２１、請求項２６、または請求項３１のいずれか１項に記載の装置において、前記剰余類計算手段における剰余の計算は、２つの各成分毎に行うことを要旨とする。

【００６０】【作用】本発明においては、代数体のイデアルの剰余類の構成法とべき乗の演算アルゴリズムを与えることにより、代数体上に拡張された公開鍵暗号方式を実現するための鍵生成方法、暗号化方法および復号化方法を具体的に構成し、また鍵生成装置、暗号化装置、および復号化装置を具体的に構成した。

【００６１】すなわち、本発明に係る公開鍵暗号による鍵生成方法および鍵生成装置によれば、従来の有理整数環上での公開鍵暗号を円分体または２次体の整数環の上へ拡張した公開鍵暗号方式を提供することができる。

【００６２】また、本発明に係る公開鍵暗号による暗号化方式、復号化方式、暗号化装置及び復号化装置によれば、円分体または２次体それぞれにおけるイデアルを法とするべき乗方法及びイデアルを法とするべき乗演算装置を提供することにより、ＲＳＡ型の公開鍵暗号方式及び暗号化通信方式及び暗号化通信装置を提供することができる。

【００６３】その結果、同報通信に対する安全性の向上したＲＳＡ型暗号を構成することができる。また、本発明による公開鍵暗号によれば、従来の楕円曲線上のＲＳＡ暗号に比べて暗号化速度を大幅に高速化することができる。

【００６４】また、本発明に係る公開鍵暗号による認証文生成方法、認証文検証方法、認証文生成装置、及び認証文検証装置によれば、円分体または２次体それぞれにおけるイデアルを法とするべき乗方法及びイデアルを法とするべき乗演算装置を提供することにより、ＲＳＡ型の公開鍵暗号方式による認証通信が行える。

【００６５】また、本発明による公開鍵暗号方式を用いた暗号化通信装置は、認証にも適用でき、暗号化通信および認証通信の双方に対称使用が可能となる。

【００６６】

【発明の実施の形態】

【本発明に係る公開鍵暗号方式の原理】次に、本発明に係る公開鍵暗号方式の原理を詳細に説明する。有理整数環Ｚにおいて、素数pと互いに素な任意の整数aに対して、

【数８】

$$a^{p-1} \equiv 1 \mod p \qquad \cdots (10)$$

式（１０）で示されるフェルマーの小定理が成り立つことが知られている。

【００６７】任意の代数体の整数環Ｏに於いても、有理整数環Ｚと同様にフェルマーの小定理が成り立つ。つまり、Ｏの素イデアルをＰとすれば、Ｐと素な元αに対して、

【数９】

$$\alpha^{Nrm P-1} \equiv 1 \mod P \quad \cdots (11)$$

式（１１）が成り立つ。ただし、Ｎｒｍはイデアルのノルムとする。

【００６８】また、素イデアルＰ，Ｑに対して、

【数１０】

$$N = PQ \quad \cdots (12)$$

$$L = LCM(NrmP-1, NrmQ-1) \quad \cdots (13)$$

とする。ここで、フェルマーの小定理より、Ｋ≡１（ｍｏｄ Ｌ）をみたすＫに対して、

【数１１】

$$\alpha^{k} \equiv \alpha \mod N \quad \cdots (14)$$

が成立する。この合同式（１４）で、Ｋ≡ｅｄ（ｍｏｄ Ｌ）を満たすｅ，ｄを選び、Ｎの剰余類を第１の公開鍵、ｅを第２の公開鍵、ｄを第２の秘密鍵とすれば、

【数１２】

$$\alpha^{ed} \equiv \alpha \mod N \quad \cdots (15)$$

合同式（１５）が成り立ち、公開鍵方式の暗号化および復号化が可能となる。本原理に基づき、本発明である代数体上に拡大したＲＳＡ型の公開鍵暗号が構成できる。

【００６９】［暗号化通信装置の実施形態］次に図面を参照して、本発明の実施の形態を詳細に説明する。図１は、本発明に係る公開鍵暗号による暗号化通信装置１１の全体構成を説明するブロック図である。図１によれば、暗号化通信装置１１は、鍵生成装置２１、暗号化装置３１、復号化装置４１、および通信路５１から構成されている。

【００７０】鍵生成装置２１は、公開鍵１（ｎ）、公開鍵２ｅ、秘密鍵１（ｐ），（ｑ）、および秘密鍵２ｄを生成する装置である。

【００７１】暗号化装置３１は、平文Ｍを暗号化して暗号文（Ｃ０，Ｃ１，…，Ｃｒ－１）を通信路５１に送出する装置であり、平文Ｍを受け入れる平文入力部３３、平文Ｍを一連の分割平文（Ｍ０，Ｍ１，…，Ｍｒ－１）に分割する平文分割部３５、および分割平文（Ｍ０，Ｍ１，…，Ｍｒ－１）を暗号化して暗号文（Ｃ０，Ｃ１，…，Ｃｒ－１）を得て、この暗号文を送信する暗号化処理部３７を備えている。

【００７２】復号化装置４１は、ｒ個のブロックからなる暗号文（Ｃ０，Ｃ１，…，Ｃｒ－１）を復号化して分割平文（Ｍ０，Ｍ１，…，Ｍｒ－１）を得る復号化処理部４３、この

$$(C0, C1, \cdots, Cr-1) \equiv (M0, M1, \cdots, Mr-1)^{e} \mod N \quad \cdots (16)$$

式（１６）によって暗号化を行なう。この暗号文は、通信路５１を介して、受け手に送られる。

【００８０】［復号化装置］復号化装置４１は、ｒ個の分割平文（Ｍ０，Ｍ１，…，Ｍｒ－１）を統合して平文Ｍを得る平文統合処理部４７、および平文Ｍを出力する平文出力部４９を備えて構成されている。通信路５１は、従来の通信路と同様の通信路である。

【００７３】次に、この暗号化通信装置１１を構成する各装置の機能の概略を説明する。

［鍵生成装置］まず、鍵生成装置２１は、素イデアル生成部２３により２個の素イデアルＰ，Ｑ（秘密鍵１）を生成し、その積Ｎ＝ＰＱの剰余類（公開鍵１）を決定する。次いで、素イデアルＰ，ＱからＬを計算し、ｅ（公開鍵２）と、ｄ（秘密鍵２）を生成するものである。

【００７４】鍵生成装置２１における鍵生成処理は、［２個の素イデアル（秘密鍵１）の生成］、［剰余類（公開鍵１）の決定］及び［公開鍵２であるｅと秘密鍵２であるｄの生成］の３段階からなり、代数体が円分体であるか、２次体であるかによって、処理内容が異なる。

【００７５】［円分体における鍵生成処理］円分体の場合は、まず、円分体を生成する原始根の位数ｍを入力として、互いに異なる２つの秘密鍵１である素イデアルＰ，Ｑを出力する。次いで、２つの秘密鍵１である素イデアルＰ，Ｑを入力として、公開鍵２であるｅと秘密鍵２であるｄを生成し出力する。

【００７６】［２次体における鍵生成処理］２次体の場合は、判別式Ｄを入力として、互いに異なる２つの秘密鍵１である素イデアルＰ，Ｑを出力する。次いで、２つの秘密鍵１である素イデアルＰ，Ｑを入力として、公開鍵２であるｅと秘密鍵２であるｄを生成し出力する。

【００７７】［暗号化装置］次に、暗号化装置３１は、平文入力部３３、平文分割処理部３５、及び暗号化処理部３７を含んで構成され、暗号化処理部３７は、イデアルｎを法とするべき乗演算部を含んでいる。

【００７８】平文入力部３３は、送信すべきメッセージである平文Ｍを受け入れる。平文分割処理部３５は、鍵生成装置２１よりブロック数ｒおよび分割平文長［ｌｏｇ２ｎ］（［ｘ］は、ｘを超えない最大の整数を示すガウスの記号である）を得て、平文を一連の分割平文（Ｍ０，Ｍ１，…，Ｍｒ－１）に分割する。ここで、ｒをイデアルＮの剰余類の完全代表系としてのベクトル空間Ｚ上の整数環Ｏの次元とする。そして、平文Ｍに対して、それぞれのブロックが剰余類の範囲内のｒ個のブロックからなる分割平文（Ｍ０，Ｍ１，…，Ｍｒ－１）を生成する。

【００７９】暗号化処理部３７は、イデアルｎを法とするべき乗演算部３９により、受け手の公開鍵ｅを用いて、

【数１３】

$$(C0, C1, \cdots, Cr-1) \equiv (M0, M1, \cdots, Mr-1)^{e} \mod N \quad \cdots (16)$$

ブロックからなる暗号文（Ｃ０，Ｃ１，…，Ｃｒ－１）を復号化して分割平文（Ｍ０，Ｍ１，…，Ｍｒ－１）を得る復号化処理部４３、この分割平文（Ｍ０，Ｍ１，…，Ｍｒ－１）を統合

して平文Mを得る平文統合処理部４７、および平文Mを出力する平文出力部４９を備えて構成されている。このうち復号化処理部４３は、イデアルｎを法とするべき乗演算部４５を備えている。

$$(C0, C1, \cdots, Cr-1)^d \equiv (M0, M1, \cdots, Mr-1) \mod N \quad \cdots (17)$$

式（１７）を計算することによって、復号化を行なう。

【００８２】平文統合処理部４７は、復号化された分割平文（M0, M1, ⋯, Mr－1）を統合し、原平文Mへ復元し、平文出力部４９から出力する。

【００８３】次に、鍵生成装置２１における本発明の公開鍵暗号方式に用いられる公開鍵（暗号化鍵）及び秘密鍵（復号化鍵）の生成方法を詳細に説明する。

【００８４】［円分体における素イデアルの生成］まず、円分体における素イデアルの生成について説明する。有理数体Qに１の原始ｍ乗根ζ_m（ただし、ｍは素数のべきとする）を添加した体Q（ζ_m）をｍ次円分体とよぶ。その整数環をOと書く。ｍ＝３，５，７，１１，１３などのときは、ユークリッド環であった。また、この円分体の整数環Oは$Z_{\phi(m)}$に埋め込み可能であり、以降Oの元

$$a0 + a1\ \zeta_m + a2\ (\zeta_m)^2 + \cdots + a\ \phi(m)-1\ (\zeta_m)^{\phi(m)-1} \quad \cdots (18)$$

を$Z_{\phi(m)}$の元（a0，a1，a2 ⋯，aϕ(m)－1）と同一視する。ただし、a0，a1，a2 ⋯，aϕ(m)－1，$\in Z$であり、ϕ（ｍ）は１以上ｍ－１以下の自然数でｍと互いに素な数の個数である（オイラー関数）。したが

$$p^{\phi(m)} \equiv 1 \pmod m, \quad p^i \not\equiv 1 \pmod m$$

であるときOにおいても素数である。つまり、上の条件を満たす素数ｐを選べば、その素数は円分体の整数環でも分解せず、素イデアル（ｐ）の生成元ｐとなる。これを、惰性する素数と呼び、暗号化装置の秘密鍵１とする。

【００８６】具体例として、ｍ＝３，５，７，１１，１３の場合などは、素数ｐとして、

【数１７】

ｍ＝３　ｐ≡２　mod ３

ｍ＝５　ｐ≡２，３　mod ５

ｍ＝７　ｐ≡３，５　mod ７

ｍ＝１１　ｐ≡２，６，７，８　mod １１

ｍ＝１３　ｐ≡２，６，７，９，１１　mod １３

を選べばよい。円分体は、原始ｍ乗根の自然数ｍが与えられると決定する。

【００８７】図３は、円分体における素イデアル生成装置の機能図である。同図によれば、まず、円分体を生成する原始根の位数ｍを入力し、次いで、有理素数ｐを任意に生成する（ステップＳ１０１）。次いで、制御変数

$$\omega = (1 + \sqrt{m}) / 2 \quad m \equiv 1 \pmod 4 \quad \cdots (20)$$

$$\omega = \sqrt{m} \quad m \equiv 2, 3 \pmod 4 \quad \cdots (21)$$

とするとき、任意の整数ａ，ｂを用いて、

【数１９】

$$\alpha = a + b\omega \quad \cdots (22)$$

と書ける。そこで、２次体の整数環の元と２次元平面Z^2を同一視する。したがって、２次体の整数環OはZ^2上２次元である。

【００９０】また、２次体の判別式Dをｍの４を法とする剰余の値に従って、

【数２０】

$$D = m \quad m \equiv 1 \pmod 4 \quad \cdots (23)$$

$$D = 4m \quad m \equiv 2, 3 \pmod 4 \quad \cdots (24)$$

と定義する。このとき、偶数でなく判別式で割れない素数ｐに対して、

【数２１】

$$(D/p)^2 = 1 \quad \cdots (25)$$

を満たす素数ｐは、平方非剰余であり、２次体の整数環Oの上でも分解せず素イデアル（ｐ）の生成元となる。ここで、

【数２２】$(D/p)^2$

は、平方剰余記号（ルジャンドルの記号とも呼ばれる）

【００９１】復号化処理部４３は、イデアルｎを法とするべき乗演算部４５により、受け手自身の秘密鍵ｄを用いて、

【数１４】

$$(C0, C1, \cdots, Cr-1)^d \equiv (M0, M1, \cdots, Mr-1) \mod N \quad \cdots (17)$$

式（１７）を計算することによって、復号化を行なう。

【００９２】平文統合処理部４７は、復号化された分割平文（M0, M1, ⋯, Mr－1）を統合し、原平文Mへ復元し、平文出力部４９から出力する。

【００９３】次に、鍵生成装置２１における本発明の公開鍵暗号方式に用いられる公開鍵（暗号化鍵）及び秘密鍵（復号化鍵）の生成方法を詳細に説明する。

【００９４】［円分体における素イデアルの生成］まず、円分体における素イデアルの生成について説明する。有理数体Qに１の原始ｍ乗根ζ_m（ただし、ｍは素数のべきとする）を添加した体Q（ζ_m）をｍ次円分体とよぶ。その整数環をOと書く。ｍ＝３，５，７，１１，１３などのときは、ユークリッド環であった。また、この円分体の整数環Oは$Z_{\phi(m)}$に埋め込み可能であり、以降Oの元

$$a0 + a1\ \zeta_m + a2\ (\zeta_m)^2 + \cdots + a\ \phi(m)-1\ (\zeta_m)^{\phi(m)-1} \quad \cdots (18)$$

って、円分体の整数環OはZ上ϕ（ｍ）次元である。

【００９５】これより、円分体の整数環Oにおいて、有理素数ｐは、

【数１６】

$$(i < \phi(m)) \quad \cdots (19)$$

ｉを１に初期設定する（ステップＳ１０３）。次いで、有理素数ｐのｉ乗（mod ｍ）を計算し、この値が１に等しいか否かを判定する（ステップＳ１０５）。

【００９８】等しければ、ステップＳ１０１に戻り、再度新たな有理素数を生成する。等しくなければ、制御変数ｉがｍのオイラー関数値ϕ（ｍ）に等しいか否かを判定する（ステップＳ１０７）。等しければ、ｐが素イデアル（ｐ）の生成元ｐとなる条件を満たしているので、素イデアル（ｐ）の探索を終了する。等しくなければ、制御変数ｉを１だけ増加させて（ステップＳ１０９）、ステップＳ１０５へ戻る。

【００９９】［２次体における素イデアルの生成］次に、２次体における素イデアルの生成について説明する。有理数体Qに、二乗の因子を持たない有理整数ｍの平方根を添加して得られる代数体Q（$\sqrt{m}$）を２次体と言う。２次体の整数環をOとするとき、Oの全ての元αは、

【数１８】

$$\omega = (1 + \sqrt{m}) / 2 \quad m \equiv 1 \pmod 4 \quad \cdots (20)$$

$$\omega = \sqrt{m} \quad m \equiv 2, 3 \pmod 4 \quad \cdots (21)$$

$$D = 4m \quad m \equiv 2, 3 \pmod 4 \quad \cdots (24)$$

と定義する。このとき、偶数でなく判別式で割れない素数ｐに対して、

【数２１】

$$(D/p)^2 = 1 \quad \cdots (25)$$

を満たす素数ｐは、平方非剰余であり、２次体の整数環Oの上でも分解せず素イデアル（ｐ）の生成元となる。ここで、

【数２２】$(D/p)^2$

は、平方剰余記号（ルジャンドルの記号とも呼ばれる）

であり、例えばユークリッドの互除法を用いて計算できる。よって、上の性質を持つ素数を秘密鍵１として利用する。この素数も惰性する素数と呼ぶ。２次体は、判別式Ｄが与えられると決定することに注意する。

【００９１】図４は、２次体における素イデアル生成装置の機能図である。同図によれば、まず、２次体を生成する判別式Ｄの値を入力し、次いで、有理素数ｐを任意に生成する（ステップＳ２０１）。次いで、平方剰余記号（Ｄ／ｐ）２を計算し、この値が－１であるか否かを判定する（ステップＳ２０３）。

【００９２】（Ｄ／ｐ）２＝－１であれば、素数ｐは、２次体の整数環Ｏの上でも分解せず素イデアル（ｐ）の生成元となるので、素イデアルｐの探索を終了する。

（Ｄ／ｐ）２＝－１でなければ、ステップＳ２０１へ戻

$$L=LCM(p^{\phi(m)-1}, q^{\phi(m)-1})$$

を計算し、２次体の場合は

$$L=LCM(p^2-1, q^2-1)$$

を求める。次に、円分体２次体ともに、

$$e\,d\equiv 1 \pmod{L}$$

を満たす、ｅ，ｄを求める。秘密鍵２はｄとし、公開鍵２はｅとする。円分体における、これらの鍵を生成する装置の機能図を図５に示す。

【００９６】［平文分割処理部］次に、図６を参照して、暗号化装置３１における平文分割処理部３５の分割機能の詳細を説明する。本発明に係る公開鍵暗号方式によれば、一回の暗号化処理の対象となる分割平文のブロック数ｒは、ｍ次円分体の場合φ（ｍ）個であり、２次体の場合は２個である。このため、平文分割処理部３５により、平文Ｍをそれぞれｒ個のブロックからなる平文ブロックに分割し、分割平文（Ｍ０，Ｍ１，…，Ｍｒ－１）として、暗号化処理部３７に供給する必要がある。

【００９７】図６によれば、平文分割処理部３５の入力として、平文入力部３３より平文Ｍのビット列が与えられる。また、鍵生成装置２１より、ブロック数ｒ、及び１ブロックの長さ（ビット数）である分割平文長［ｌｏｇ２ｎ］が与えられる。分割平文長［ｌｏｇ２ｎ］は、ｌｏｇ２ｎを超えない最大の整数とする（ここで、前記２個の惰性する素数の積ｎ＝ｐｑ）。以上が平文分割処理部３５の初期状態であり、例えばメモリ上に各数値およびデータを記憶して引き渡される。

【００９８】次いで、制御変数ｉを０に設定し（ステップＳ４０１）、Ｍの先頭より［ｌｏｇ２ｎ］ビットの長

$$(C0, C1, \cdots, Cr-1) = (M0, M1, \cdots, Mr-1)^e \pmod{N} \quad \cdots (29)$$

次に、（Ｃ０，Ｃ１，…，Ｃｒ－１）を分割暗号文とし、受け手に送信する。ただし、ｒは整数環Ｏの２上の次元とする。以下で、剰余類の決定とイデアルを法とするべき乗演算処理について詳述する。

【０１０３】［イデアルを法とする剰余類の決定］本発明では、暗号化・復号化を一意的に行なうため、イデアルを法とする剰余類の取り方を以下のように定める。

り、異なる有理素数ｐを任意に生成する。

【００９３】円分体、２次体ともに、以上の方法で生成された素イデアル（ｐ）は以下生成元と同一視し、場合によりｐと記述する。

【００９４】［秘密鍵１および公開鍵１の生成］素イデアル（ｐ）が探索された方法と同様の方法により、素イデアル（ｑ）も探索されるが、素イデアル（ｐ）と異なる素イデアル（ｑ）を探索する必要がある。秘密鍵１は、これら２個の惰性する素数ｐ，ｑとする。公開鍵１は、それらの積ｎ＝ｐｑとする。

【００９５】［秘密鍵２および公開鍵２の生成］秘密鍵１ｐ，ｑに対して、ＬＣＭを最小公倍数を求める関数とするとき、ｍ次円分体の場合は、

【数２３】

$$\cdots (26)$$

【数２４】

$$\cdots (27)$$

【数２５】

$$\cdots (28)$$

さのブロックを切り出し、この切り出されたブロックをＭｉとし、残りのビット列を新たなＭとする（ステップＳ４０３）。ブロックＭｉは、例えばメモリ上に確保されたｒの配列記憶位置のｉ番目に格納してもよい。

【００９９】次いで、制御変数ｉがｒ－１に等しいか否かを判定し（ステップＳ４０５）、この判定でｉ≠ｒ－１であれば、ｉを１だけ増加させて（ステップＳ４０７）、残りのブロックの切り出しのために、ステップＳ４０３へ戻る。

【０１００】ステップＳ４０５の判定において、ｉ＝ｒ－１であれば、暗号化単位のｒブロックの切り出しが終わったので、Ｍ０，Ｍ１，…，Ｍｒ－１を暗号化処理部３７へ出力する（ステップＳ４０９）。次いで、Ｍが空かどうかを判定し（ステップＳ４１１）、Ｍが空であれば、平文分割処理を終了し、Ｍが空でなければ、再度ｒ個のブロックを切り出すために、ステップＳ４０１へ戻る。以上のようにして、平文Ｍがブロック分割される。

【０１０１】［暗号化処理部］次に、暗号化処理部３７の詳細を説明する。平文分割処理部３５から入力された分割平文（Ｍ０，Ｍ１，…，Ｍｒ－１）に対して、イデアルＮを法とする以下のべき乗演算を行なう。

【０１０２】

【数２６】

$$(C0, C1, \cdots, Cr-1) = (M0, M1, \cdots, Mr-1)^e \pmod{N} \quad \cdots (29)$$

【０１０４】［円分体における剰余類］まず、円分体におけるイデアルを法とする剰余類の取り方を説明する。ユークリッド的なｍ次円分体において０でないイデアルＮによる剰余類の完全代表系として、Ｚφ(ｍ)に於いて原点からベクトル

【数２７】Ｎ，τＮ，τ２Ｎ，…，τφ(ｍ)－１Ｎ

で張られる超平行四辺体の内部と境界上の格子点が取れ

る。ただし、境界上の格子点は原点に面さない超平面上の点は除く。

【0105】また、任意の円分体において、惰性する素数の核で生成されるイデアル（n）による剰余類の完全代表系としてZφ(m)に於いて、原点と

【数28】 n，τn，τ2 n，…，τφ(m)-1n

を頂点とする超立方体の内部と境界上の格子点が取れる。ただし、各軸を一辺としない超平面上の点は除く。

【0106】ここで、注意として惰性する素数の核で生成されるイデアル（n）による剰余類の計算方法は、各成分で有理整数の意味で（mod n）を計算することである。

【0107】［2次体における剰余類］次に、2次体におけるイデアルを法とする剰余類の取り方を説明する。ユークリッド的な2次体において0でないイデアルNによる剰余類の完全代表系として、Z2に於いて、原点からベクトル

【数29】 N，ωN

で張られる平行四辺形の内部と境界上の格子点が取れる。ただし、境界上の格子点は原点に面する超平面上の点のみ含める。

【0108】また、任意の2次体において、惰性する素数の核で生成されるイデアル（n）による剰余類の完全代表系として、2次元平面Z2に於いて、原点と、

【数30】 n，ωn

を頂点とする正方形の内部と、境界上の格子点（各座標が整数値となる点）が取れる。ただし、各軸を一辺としない面上の点は除く。

【0109】ここで、注意として惰性する素数の核で生成されるイデアル（n）による剰余類の計算方法は、各成分毎に有理整数の意味で（mod n）を計算することである。

【0110】また、任意の2次体において、惰性する素数の核で生成されるイデアル（n）による剰余類の完全代表系として、2次元平面Z2（各軸を次元1・次元2と呼ぶ）に於いて、原点（0，0）と、（n，0）、（n，n）、（0，n）を頂点とする正方形の内部と、境界上の格子点（各座標が整数値となる点）が取れる。ただし、各軸を一辺としない面上の点は除く。この様子を図32に示す。

【0111】［イデアル（n）を法とするべき乗演算］次に、イデアル（n）を法とするべき乗演算について説明する。本発明の公開鍵暗号方式では、式（16）、（17）に示したようなべき乗演算が行われる。べき乗演算では、指数部を2進表現して、この関数を指数部が2のべきとなる因数の核に分解し、小さい因数から順次、法による乗算を繰り返していく、反復平方核によるべき乗演算法、または2進計算法と呼ばれる既知の高速べき乗演算方法を使用することができる。

【0112】このべき乗計算のための基本演算は、

【数31】

$$x2n=(xn)2,\quad x2n+1=(xn)2\times x \qquad \cdots(30)$$

式（30）に示すように、乗算と2乗演算であるので、この乗算と2乗演算方法を以下に説明する。

【0113】［円分体における乗算および2乗演算方法］まず、円分体における乗算および2乗演算方法について説明する。円分体に於ける2元x＝（x0，x1，…xφ(m-1)）とy＝（y0，y1，…，yφ(m)-1）

$$x(\times)y=(x(\times)y0,\ x(\times)y1,\ \cdots,\ x(\times)y\varphi(m)-1) \qquad \cdots(31)$$

$$x(\times)x=(x(\times)x0,\ x(\times)x1,\ \cdots;\ x(\times)x\varphi(m)-1) \qquad \cdots(32)$$

ここで、イデアル（n）の剰余の取り方は、すべての成分の式を（mod n）で計算する。これにより、イデアル（n）を法とするべき乗演算装置が構成できる。

【0115】次に、円分体に於いて、拡大次数が低次でかつ素数の場合のアルゴリズムを詳細に記述する。特に、m＝3の場合のべき乗演算を行なうための乗算と2乗演算装置を図7と図8に示す。図7及び図8において、符号301、303、313、315、325、3

$$x(\times)y=(x(\times)y0,\ x(\times)y1) \qquad \cdots(33)$$

ここで、

$$x(\times)y0=x0\,y0-x1\,y1$$
$$x(\times)y1=x1\,y0+(x0-x1)\,y1$$
$$x(\times)x=(x(\times)x0,\ x(\times)x1) \qquad \cdots(34)$$

31及び333は、通常の乗算回路であり、符号305、311、323は減算回路、符号317及び321は加算回路、符号307、319、327、335は、mod n演算回路である。

【0116】［m＝3の場合の乗算と2乗演算］m＝3の場合、オイラー関数値φ(m)は、φ(3)＝2となり、乗算および2乗演算は、

【数33】

ここで、

$x(x)\,x0 = (x0 + x1)(x0 - x1)$

$x(x)\,x1 = 2 \cdot x0 \cdot x1$

それぞれ式（３３）、（３４）に示すとおりとなる。

$$x(x)\,y = (x(x)\,y0,\ x(x)\,y1,\ \cdots,\ x(x)\,y3) \quad \cdots(35)$$

ここで、

$x(x)\,y0$
$= x3\,(y2 - y1) + x2\,(y3 - y2) + x0\,y0 - x1\,y3$

$x(x)\,y1$
$= x1\,(y0 - y3) + (x0 + x3)\,y1 - x2\,y2 - x3\,y3$

$x(x)\,y2$
$= x2\,(y0 - y2) + x1\,(y1 - y3) + x0\,y2 - x3\,y1$

$x(x)\,y3$
$= x3\,(y0 - y1) + x2\,(y1 - y2) + x2\,y1 - x1\,y3$

$$x(x)\,x = (x(x)\,x0,\ x(x)\,x1,\ \cdots,\ x(x)\,x3) \quad \cdots(36)$$

ここで、

$x(x)\,x0 = (x0 + x2)(x0 - x2) + 2x3\,(x2 - x1)$

$x(x)\,x1 = 2x1\,(x0 - x3) + (x3 + x2)\,(x3 - x2)$

$x(x)\,x2 = x1\,(x1 - 2x3) + x2\,(2x0 - x2)$

$$x(x)\,y = (x(x)\,y0,\ x(x)\,y1,\ \cdots,\ x(x)\,y5) \quad \cdots(37)$$

ここで、

$x(x)\,y0 = x0\,y0 - x5\,y1 + (-x4 + x5)\,y2 + (-x3 + x4)\,y3 + (-x2 + x3)\,y4 + (-x1 + x2)\,y5$

$x(x)\,y1 = x1\,y0 - x4\,y2 + (x0 - x5)\,y1 + (-x3 + x5)\,y3 + (-x2 + x4)\,y4 + (-x1 + x3)\,y5$

$x(x)\,y2 = x2\,y0 - x3\,y3 + (x1 - x5)\,y1 + (x0 - x4)\,y2 + (-x2 + x5)\,y4 + (-x1 + 2x4)\,y5$

$$x(x)\,x = (x(x)\,x0,\ x(x)\,x1,\ \cdots,\ x(x)\,x5) \quad \cdots(38)$$

ここで、

$x(x)\,x0 = (x0 + x3)(x1 - x3) + 2x4\,(x3 + x2) + 2x5\,(x2 - x1)$

$x(x)\,x1 = 2x1\,(x0 - x5) + x3\,(2x5 - x3) + x4\,(x4 - 2x2)$

$x(x)\,x2 = (x1 + x3)(x1 - x3) + 2x5\,(x4 - x1) + x2\,(x0 - x4)$

$x(x)\,x3 = x5\,(x5 - 2x1) + x3\,(2x0 - x3) + 2x2\,(x1 - x4)$

$$x(x)\,y = (x(x)\,y0,\ x(x)\,y1,\ \cdots,\ x(x)\,y9) \quad \cdots(39)$$

ここで、

$x(x)\,y0 = x0\,y0 - x9\,y1 + (-x8 + y9)\,y2 + (-x7 + x8)\,y3 + (-x6 + x7)\,y4 + (-x5 + x6)\,y5 + (-x4 + x5)\,y6 + (-x3 + x4)\,y7 + (-x2 + x3)\,y8 + (-x1 + x2)\,y9$

【０１１７】［m＝５の場合の乗算と２乗演算］m＝５の場合、オイラー関数値φ（m）は、φ（５）＝４となり、乗算および２乗演算は、

【数３４】

【０１１８】［m＝７の場合の乗算と２乗演算］m＝７の場合、オイラー関数値φ（m）は、φ（７）＝６となり、乗算および２乗演算は、

【数３５】

【数３６】

【０１１９】［m＝１１の場合の乗算と２乗演算］m＝１１の場合、オイラー関数値φ（m）は、φ（１１）＝１０となり、乗算および２乗演算は、

【数３７】

$x(x)\,y1 = x1\,y0 - x8\,y2 + (x0 - x9)\,y1 + (-x7 + x9)\,y3 + (-x6 + x8)\,y4 + (-x5 + x7)\,y5 + (-x4 + x6)\,y6 + (-x3 + x5)\,y7 + (-x2 + x4)\,y8 + (-x1 + x3)\,y9$

$x(x)\,y2 = x2\,y0 - x7\,y3 + (x1 - x9)\,y$

1 + (x0 − x8) y2 + (− x6 + x9) y4 + (− x5 + x8) y5 + (− x4 + x7) y6 + (− x3 + x6) y7 + (− x2 + x5) y8 + (− x1 + x4) y9

x (x) y3 = x3 y0 − x6 y4 + (x2 − x9) y1 + (x1 − x8) y2 + (x0 − x7) y3 + (− x5 + x9) y5 + (− x4 + x8) y6 + (− x3 + x7) y7 + (− x2 + x6) y8 + (− x1 + x5) y9

x (x) y4 = x4 y0 − x5 y5 + (x3 − x9) y1 + (x2 − x8) y2 + (x1 − x7) y3 + (x0 − x6) y4 + (− x4 + x9) y6 + (− x3 + x8) y7 + (− x2 + x7) y8 + (− x1 + x6) y9

x (x) y5 = x5 y0 − x4 y6 + (x4 − x9) y1 + (x3 − x8) y2 + (x2 − x7) y3 + (x1 − x6) y4 + (x0 − x5) y5 + (− x3 + x9) y7 + (− x2 + x8) y8 + (− x1 + x7) y9

x (x) y6 = x6 y0 − x3 y7 + (x5 − x9) y1 + (x4 − x8) y2 + (x3 − x7) y3 + (x2 − x6) y4 + (x1 − x5) y5 + (x0 − x4) y6 + (− x2 + x9) y8 + (− x1 + x8) y9

x (x) y7 = x7 y0 − x2 y8 + (x6 − x9) y1 + (x5 − x8) y2 + (x4 − x7) y3 + (x3 − x6) y4 + (x2 − x5) y5 + (x1 − x4) y6 + (x0 − x3) y7 + (− x1 + x9) y9

x (x) y8 = x8 y0 − x1 y9 + (x7 − x9) y1 + (x6 − x8) y2 + (x5 − x7) y3 + (x4 − x6) y4 + (x3 − x5) y5 + (x2 − x4) y6 + (x1 − 8x3) y7 + (x0 − x2) y8

x (x) y9 = x9 y0 + (x8 − x9) y1 + (x7 − x8) y2 + (x6 − x7) y3 + (x5 − x6) y4 + (x4 − x5) y5 + (x3 − x4) y6 + (x2 − x3) y7 + (x1 − x2) y8 + (x0 − x1) y9

【数３８】

x (x) = (x (x) x0 , x (x) x1 , ⋯, x (x) x9) ⋯ (40)

ここで、

x (x) x0 = (x0 + x5)(x0 − x5) + 2x9 (x2 − x1) + 2x8 (x3 − x2) + 2x7 (x4 − x3) + 2x7 (x5 − x4)

x (x) x1 = 2x1 (x0 − x9) + 2x3 (x9 − x7) + 2x8 (x4 − x2) + x6 (x6 − 2x4) + x5 (2x7 − x5)

x (x) x2 = 2x2 (x0 − x8) + 2x4 (x9 − x6) + x1 (x1 − 2x8) + x5 (2x8 − x5) + 2x7 (x6 − x3)

x (x) x3 = 2x3 (x0 − x7) + 2x2 (x1 − x8) + 2x9 (x5 − x1) + 2x6 (x8 − x4) + (x9 + x5)(x9 − x5)

x (x) x4 = 2x4 (x0 − x6) + 2x3 (x1 − x7) + 2x9 (x6 − x1) + 2x8 (x7 − x2) + (x2 + x5)(x2 − x5)

x (x) x5 = 2x5 (x0 − x5) + 2x1 (x4 − x6) + 2x2 (x3 − x8) + 2x9 (x7 − x1)

x (x) y = (x (x) y0 , x (x) y1 , ⋯, x (x) y11) ⋯ (41)

ここで、

x (x) y0 = x0 y0 − x11 y1 + (− x2 + x3) y10 + (− x1 + x2) y11 + (− x10 + x11) y2 + (x10 − x9) y3 + (− x8 + x9) y4 + (− x7 + x8) y5 + (− x6 + x7) y6 + (− x5 + x6) y7 + (− x4 + x5) y8 + (− x3 + x4) y9

x (x) y1 = x1 y0 − x10 y2 + (x0 − x11) y1 + (− x2 + x4) y10 + (− x1 + x3) y11 + (x11 − y9) y3 + (x10 − x8) y4 + (− x7 + x9) y5 + (− x6 + x8) y6 + (− x5 + x7) y7 + (− x4 + x6) y8 + (− x3 + x5) y9

x (x) y2 = x2 y0 − x9 y3 + (x1 − x11) y1 + (− x2 + x5) y10 + (− x1 + x4) y11 + (x0 − x10) y2 + (x11 − x8) y4 + (x10 − x7) y5 + (x9 − x6) y6 + (x8 − x5) y7 + (x7 − x4) y8 + (x6 − x3) y9

x (x) y3 = x3 y0 − x8 y4 + (− x11 + x2) y1 + (− x2 + x6) y10 + (− x1 + x5) y11 + (x1 − x10) y2 + (x0 − x9) y3 + (x11 − x7) y5 + (x10 − x6) y6 + (− x5 + x9) y7 + (− x4 + x8) y8 + (− x3 + x7) y9

x (x) y4 = x4 y0 − x7 y5 + (− x11 + x3) y1 + (− x2 + x7) y10 + (− x1 + x6) y11 + (− x10 + x2) y2 + (x1 − x9) y3 + (x0 − x8) y4 + (x11 − x6) y6 + (x10 − x5) y7 + (− x4 + x9) y8 + (− x3 + x8) y9

x (x) y5 = x5 y0 − x6 y6 + (− x11 + x4) y1 + (− x2 + x8) y10 + (− x1 + x7) y11 + (x10 − x3) y2 + (x9 − x4) y3 + (x8 − x7) y4 + (x11 − x9) y7 + (− x6 + x10) y8 + (− x5 + x9) y9

x (x) y6 = x6 y0 − x5 y7 + (− x11 + x5) y1 + (− x2 + x9) y10 + (− x1 + x8) y11 + (x10 − x4) y2 + (x9 − x3) y3 + (x8 − x2) y4 + (x7 − x1) y5 + (x11 − x10) y8 + (x10 − x9) y9

x (x) y7 = x7 y0 − x4 y8 + (− x11 + x6) y1 + (− x2 + x10) y10 + (− x1 + x9) y11 + (x10 − x5) y2 + (x9 − x2) y3 + (x8 − x1) y5 + (x11 − x8) y6 + (x10 − x7) y7 + (x9 − x6) y8 + (x8 − x5) y9

x (x) y8 = x8 y0 − x3 y9 + (− x11 + x7) y1 + (− x2 + x11) y10 + (− x1 + x10) y11 + (x10 − x6) y2 + (x9 − x1) y4 + (x11 − x9) y5 + (x10 − x8) y7 + (x9 − x7) y8 + (x8 − x6) y9

x (x) y9 = x9 y0 − x2 y10 + (− x11 + x8) y1 + (− x1 + x11) y2 + (x10 − x7) y3 + (x9 − x5) y4 + (x8 − x4) y5 + (x11 − x10) y6 + (x10 − x9) y8 + (x9 − x8) y9

x (x) y10 = x10 y0 − x1 y11 + (− x11 + x9) y1 + (x10 − x2) y2 + (x9 − x11) y3 + (x8 − x3) y4 + (x7 − x2) y5 + (x6 − x1) y6 + (x11 − x5) y7 + (x10 − x4) y8 + (x9 − x3) y9

x (x) y11 = x11 y0 − x10 y1 + (x11 − x1) y2 + (x10 − x11) y3 + (x9 − x10) y4 + (x8 − x9) y5 + (x7 − x8) y6 + (x6 − x7) y7 + (x5 − x6) y8 + (x4 − x5) y9

【０１２０】［m= １３の場合の乗算と２乗演算］ m= １３の場合、オイラー関数値φ (m) は、φ (１３) = １２となり、乗算および２乗演算は、

【数３９】

x (x) = (x (x) x0 , x (x) x1 , ⋯, x (x) x11) ⋯ (41)

それぞれ式 (３９) 、 (４０) に示すとおりとなる。

x (x) x0 = (x0 + x5)(x0 − x5) + 2x9 (x2 − x1) + 2x8 (x3 − x2) + 2x7 (x4 − x3) + 2x7 (x5 − x4)

x (x) x1 = 2x1 (x0 − x9) + 2x3 (x9 − x7) + 2x8 (x4 − x2) + x6 (x6 − 2x4) + x5 (2x7 − x5)

x (x) x2 = 2x2 (x0 − x8) + 2x4 (x9 − x6) + x1 (x1 − 2x8) + x5 (2x8 − x5) + 2x7 (x6 − x3)

x (x) x3 = 2x3 (x0 − x7) + 2x2 (x1 − x8) + 2x9 (x5 − x1) + 2x6 (x8 − x4) + (x9 + x5)(x9 − x5)

x (x) x4 = 2x4 (x0 − x6) + 2x3 (x1 − x7) + 2x9 (x6 − x1) + 2x8 (x7 − x2) + (x2 + x5)(x2 − x5)

x (x) x5 = 2x5 (x0 − x5) + 2x1 (x4 − x6) + 2x2 (x3 − x8) + 2x9 (x7 − x1)

x (x) y = (x (x) y0 , x (x) y1 , ⋯, x (x) y11) ⋯ (41)

ここで、

x (x) y0 = x0 y0 − x11 y1 + (− x2 + x3) y10 + (− x1 + x2) y11 + (− x10 + x11) y2 + (x10 − x9) y3 + (− x8 + x9) y4 + (− x7 + x8) y5 + (− x6 + x7) y6 + (− x5 + x6) y7 + (− x4 + x5) y8 + (− x3 + x4) y9

x (x) y1 = x1 y0 − x10 y2 + (x0 − x11) y1 + (− x2 + x4) y10 + (− x1 + x3) y11 + (x11 − y9) y3 + (x10 − x8) y4 + (− x7 + x9) y5 + (− x6 + x8) y6 + (− x5 + x7) y7 + (− x4 + x6) y8 + (− x3 + x5) y9

x (x) y2 = x2 y0 − x9 y3 + (x1 − x11) y1 + (− x2 + x5) y10 + (− x1 + x4) y11 + (x0 − x10) y2 + (x11 − x8) y4 + (x10 − x7) y5 + (x9 − x6) y6 + (x8 − x5) y7 + (x7 − x4) y8 + (x6 − x3) y9

x (x) y3 = x3 y0 − x8 y4 + (− x11 + x2) y1 + (− x2 + x6) y10 + (− x1 + x5) y11 + (x1 − x10) y2 + (x0 − x9) y3 + (x11 − x7) y5 + (x10 − x6) y6 + (− x5 + x9) y7 + (− x4 + x8) y8 + (− x3 + x7) y9

x (x) y4 = x4 y0 − x7 y5 + (− x11 + x3) y1 + (− x2 + x7) y10 + (− x1 + x6) y11 + (− x10 + x2) y2 + (x1 − x9) y3 + (x0 − x8) y4 + (x11 − x6) y6 + (x10 − x5) y7 + (− x4 + x9) y8 + (− x3 + x8) y9

x (x) y5 = x5 y0 − x6 y6 + (− x11 + x4) y1 + (− x2 + x8) y10 + (− x1 + x7) y11 +

(－x10＋x3) y2 ＋ (x2 －x9) y3＋ (x1 －x8) y4 ＋ (x0 －x7) y5＋ (x11－x5) y7 ＋ (x10－x4) y8＋ (－x3 ＋x9) y9

x (x) y6 ＝x6 y0 －x5 y7 ＋ (－x11＋x5) y1＋ (－x2 ＋x9) y10＋ (－x1 ＋x8) y11＋ (－x10＋x4) y2 ＋ (x3 －x9) y3＋ (x2 －x8) y4 ＋ (x1 －x7) y5＋ (x0 －x6) y6 ＋ (x11－x4) y8＋ (x11－x4) y8 ＋ (x10－x3) y9

x (x) y7 ＝x7 y0 －x4 y8 ＋ (－x11＋x6) y1＋ (x10－x2) y10＋ (－x1 ＋x9) y11＋ (－x10＋x5) y2 ＋ (x4 －x9) y3＋ (x3 －x8) y4 ＋ (x2 －x7) y5＋ (x1 －x6) y6 ＋ (x0 －x5) y7＋ (x11－x3) y9

x (x) y8 ＝x8 y0 －x3 y9 ＋ (－x11＋x7) y1＋ (x11－x2) y10＋ (－x1 ＋x10) y11＋ (－x10＋x6) y2 ＋ (x5 －x9) y3＋ (x4 －x8) y4 ＋ (x3 －x7) y5＋ (x2 －x6) y6 ＋ (x1 －x5) y7＋ (x0 －x4) y8 x (x) y9 ＝x9 y0 －x2 y10＋ (－x11＋x8) y1＋ (－x1 ＋x11) y11＋ (－x10＋x7) y2＋ (x6 －x9) y3 ＋ (x5 －x8) y4＋ (x4 －x7) y5 ＋ (x3 －x6) y6＋ (x2 －x5) y7＋ (x1 －x4) y8 ＋ (x0 －x3) y9

x (x) y10＝x10 y0 －x1 y11＋ (－x11＋x9) y1＋ (x0 －x2) y10＋ (－x10＋x8) y2＋ (x7 －x9) y3 ＋ (x6 －x8) y4＋ (x5 －x7) y5 ＋ (x4 －x6) y6＋ (x3 －x5) y7＋ (x2 －x4) y8＋ (x1 －x3) y9

x (x) y11＝x11 y0 ＋ (x10－x11) y1 ＋ (x1 －x2) y10＋ (x0 －x1) y11＋ (－x10＋x9) y2＋ (x8 －x9) y3 ＋ (x7 －x8) y4＋ (x6 －x7) y5 ＋ (x5 －x6) y6＋ (x4 －x5) y7 ＋ (x3 －x4) y8＋ (x2 －x3) y9

【数４０】

x (x) x＝ (x (x) x0 , x (x) x1 , ..., x (x) x11) … (42)

ここで

x (x) x0＝ (x0 ＋x6) (x0 －x6) ＋2x11 (x2 －x1) ＋2x10 (x3 －x2) ＋2x9 (x4 －x3) ＋2x5 (x8 －x7) ＋2x7 (x6 －x5)

x (x) x1＝2x1 (x0 －x11) ＋2x3 (x11－x9) ＋2x4 (x10－x8) ＋2x5 (x9 －x3) ＋x6 (2x8 －x6) ＋x7 (x7 －2x5)

x (x) x2＝2x2 (x0 －x10) ＋x1 (x1 －2x11) ＋2x4 (x11－x8) ＋2x5 (x10－x7) ＋2x6 (x9 －x6) ＋2x7 (x8 －x5)

x (x) x3＝2x3 (x0 －x9) ＋2x11 (x5 －x1) ＋2x2 (x1 －x10) ＋x6 (2x10－x6) ＋2x7 (x9 －x5) ＋x8 (x8 －2x4)

x (x) x4＝ (x2 ＋x6) (x2 －x6) ＋2x7 (x10－x5) ＋2x11 (x6 －x1) ＋x2 (x2 －2x10) ＋2x3 (x1 －x9) ＋2x4 (x0 －x8)

x (x) x5＝2x5 (x0 －x7) ＋2x1 (x4 －x11) ＋2x2 (x3 －x4) ＋2x7 (x11－x5) ＋2x10 (x8 －x2) ＋x9 (x9 －2x3)

x (x) x6＝x6 (2x0 －x6) ＋2x1 (x5 －x11) ＋2x2 (x4 －x10) ＋x3 (x3 －2x9) ＋2x8 (x11－x4) ＋2x9 (x10－x3)

x (x) x7＝2x7 (x0 －x5) ＋2x1 (x6 －x11) ＋2x2 (x5 －x10) ＋2x3 (x4 －x9) ＋2x8 (x10－x3) ＋2x9 (x11－x2)

x (x) x8＝2x8 (x0 －x4) ＋2x3 (x7 －x11) ＋2x2 (x6 －x10) ＋2x3 (x5 －x9) ＋x4 (x4 －2x8) ＋2x10 (x11－x2)

x (x) x9＝2x9 (x0 －x3) ＋2x1 (x8 －x11) ＋2x2 (x7 －x10) ＋2x3 (x6 －x9) ＋2x4 (x5 －x8) ＋x11 (x11－2x1)

x (x) x10＝2x10 (x0 －x2) ＋2x1 (x9 －x11) ＋2x2 (x8 －x10) ＋2x3 (x7 －x9) ＋2x4 (x6 －x8) ＋x5 (x5 －2x7)

x (x) x11＝2x11 (x0 －x1) ＋2x10 (x1 －x2) ＋2x9 (x2 －x3) ＋2x3 (x8 －x9) ＋2x7 (x4 －x5) ＋x6 (2x5 －x6)

それぞれ式（４１）、（４２）に示すとおりとなる。

【０１２１】【２次体における乗算および２乗演算方法】次に、２次体におけるイデアル（ｎ）の剰余の取り方を説明する。判別式がＤである、２次体の任意の２元x＝（x0，x1），y＝（y0，y1）に対する乗算x (x) y＝（x (x) y0，x (x) y1）と、任意の元のx＝（x0，x1）の２乗演算x (x) x＝（x (x) x0，x (x) x1）を行なう方法を記述する。

ここで、イデアル（ｎ）の剰余の取り方は、すべての成分の式を（mod n）で計算する。これにより、イデアル（ｎ）を法とするべき乗演算装置が構成できる。

【０１２２】まず、m≡1 (mod 4) の場合

【数４１】

x (x) y0＝x0 y0＋ (m－1) (1／4) x1 y1
x (x) y1＝x1 (y0＋ (m－1) (1／4) y1) ＋x0 y1
x (x) x0＝ (x0) 2＋ (m－1) (1／4) (x1) 2
x (x) x1＝x1 (x0＋ (m－1) (1／4) x1) … (43)

【０１２３】次いで、m≡2，3 (mod 4) の場合

式（４３）となる。

【数４２】

$$x\ (x)\ y0 = x0\ y0 + m x1\ y1$$
$$x\ (x)\ y1 = x1\ (y0 + m x1) + x0\ y1$$
$$x\ (x)\ x0 = (x0)^2 + m (x1)^2$$
$$x\ (x)\ x1 = x1\ (x0 + m x1) \quad \cdots (44)$$

式（４４）となる。

【０１２４】［復号化処理部］次に、復号化装置４１の復号化処理部４３の動作を説明する。復号化処理部４３は、通信路５１を介して暗号化装置３１より送られてきた分割暗号文（C0，C1，…，Cr-1）に対して、

【数４３】

$$(C0, C1, \cdots, Cr-1)^d \equiv (M0, M1, \cdots, Mr-1) \mod (n) \quad \cdots (45)$$

式（４５）を計算することによって、分割平文（M0，M1，…，Mr-1）を復元する。

【０１２５】復号化処理部４３のイデアル（n）を法とするべき乗演算部４５は、秘密鍵２ｅによりべき乗演算を行うこと以外は、暗号化装置３１のイデアル（n）を法とするべき乗演算部３７と同じであり、暗号化装置３１と復号化装置４１の大部分は共通であり、暗号化・復号化装置として構成することにより、双方向の暗号化通信を行うことができる。

【０１２６】［平文統合処理部］平文統合処理部４７は、復号処理部によって得られる分割平文（M0，M1，…，Mr-1）を順番に連結し原平文Mを得る。

【０１２７】次に、フローチャート図およびメモリ上のデータ配置を示す表を参照しながら、本発明の実施形態を詳細に説明する。図９ないし図１２は、円分体暗号における鍵生成の詳細手順を示すフローチャート図である。

【０１２８】円分体暗号における鍵生成では、表１に示すようなメモリ上のデータ配置を行う。

【０１２９】

【表１】

データ番号	データ内容
1	円分体の原始根の位数m
2	オイラー関数値φ（m）
3	有限素数 p
4	素イデアル１（秘密鍵１） p
5	素イデアル２（秘密鍵１） q
6	公開鍵 n = pq
7	最小公倍数 L
8	秘密鍵２ d
9	公開鍵２ e

そして、図９に示すように、まず、円分体を生成する原始根の位数mをデータ番号１に入力する（ステップＳ５０１）。次いで、オイラー関数値φ（m）を計算し（ステップＳ５１０）、オイラー関数値φ（m）をデータ番号２に格納する（ステップＳ５３１）。次いで、素イデアル１（秘密鍵１）pを生成し（ステップＳ５４０）、素イデアル２（秘密鍵１）qを生成し（ステップＳ５６０）、素イデアルp，qからその積である公開鍵１，n=pqを求めて、データ番号６に格納する（ステップＳ５８１）。

【０１３０】次いで、pφ(m)－１とqφ(m)－１との最小公倍数であるLを計算し、これをデータ番号７に格納する（ステップＳ５８３）。次いで、ed≡１（mod L）となる２数、eおよびdを求め、秘密鍵２dをデータ番号８に、公開鍵２eをデータ番号９に、それぞれ格納して（ステップＳ５８５）鍵生成を終了する。

【０１３１】図１０は、オイラー関数値φ（m）の計算ルーチンを示すフローチャートであり、ステップＳ５１０以下の処理の詳細を示す。まず、mを入力し（ステップＳ５１２）、次いで制御変数i，jをそれぞれ、１，０に初期設定する（ステップＳ５１４）。次いで、iとmとの公約数が存在するか否かを判定する（ステップＳ５１６）。ステップＳ５１６の判定において、yesであれば（公約数が存在すれば）なにもせず、noであれば（公約数が存在しなければ）jを１だけ増加させて（ステップＳ５１８）、ステップＳ５２０へ移る。

【０１３２】次いで、iを１だけ増加させて（ステップＳ５２０）、iとmとが等しいか否かを判定する（ステップＳ５２２）。ステップＳ５２２の判定において、iとmとが等しくなければ、ステップＳ５１６へ戻り、iとmとが等しければ、φ（m）＝jとして計算を終了し（ステップＳ５２４）、次の処理へ移る。

【０１３３】図１１は、素イデアル１，pの生成処理を示すフローチャートであり、ステップＳ５４０以下の処理の詳細を示す。まず、所定の範囲の有理整数pを任意に（ランダムに）生成し、データ番号３に格納する（ステップＳ５４２）。次いで、制御変数iを１に初期設定する（ステップＳ５４４）。次いで、pi （mod m）を計算し、この値が１であるか否かを判定する（ステップＳ５４６）。

【０１３４】ステップＳ５４６の判定で、pi ＝１（mod m）であれば、ステップＳ５４２に戻り、再度有理素数pを生成する。pi ＝１（mod m）でなければ、i＝φ（m）か否かを判定する（ステップＳ５４８）。ステップＳ５４８の判定において、i＝φ（m）でなければ、iに１を加えて（ステップＳ５５０）、ステップＳ５４６へ戻る。i＝φ（m）であれば、pは素イデアルであると判定できるので、素イデアル１（秘密

鍵1）pをデータ番号4に格納して（ステップS552）、次の処理へ移る。

【0135】図12は、素イデアル2，qの生成処理を示すフローチャートであり、ステップS560以下の処理の詳細を示す。まず、所定の範囲の有理整数qを任意に（ランダムに）生成する（ステップS562）。次いで、qがpに等しいか否かを判定する（ステップS564）。qとpとが等しければ、pと異なる有理整数qを発生させるために、ステップS562へ戻る。qがpと異なれば、次いで、制御変数iを1に初期設定する（ステップS566）。次いで、q^i（mod. m）を計算し、この値が1であるか否かを判定する（ステップS568）。

【0136】ステップS568の判定で、q^i=1（mod m）であれば、ステップS562に戻り、再度有理素数qを生成する。q^i=1（mod m）でなければ、i=φ（m）か否かを判定する（ステップS570）。ステップS570の判定において、i=φ（m）でなければ、iに1を加えて（ステップS572）、ステップS568へ戻る。i=φ（m）であれば、qは素イデアルであると判定できるので、素イデアル2（秘密鍵1）qをデータ番号5に格納して（ステップS574）、次の処理へ移る。

【0137】図13ないし図16は、2次体暗号における鍵生成の詳細手順を示すフローチャートである。

【0138】2次体暗号における鍵生成では、表2に示すようなメモリ上のデータ配置を行う。

【0139】

【表2】

データ番号	データの内容
1	2乗因子を持たない有理整数m
2	判別式D
3	有理素数p
4	素イデアル1（秘密鍵1）p
5	素イデアル2（秘密鍵2）q
6	公開鍵1 n=p·q
7	最小公倍数 L
8	秘密鍵2 d
9	公開鍵2 e

そして、図13に示すように、まず2乗因子を持たない有理整数mをデータ番号1に入力する（ステップS601）。次いで、判別式Dの値をデータ番号2に格納する（ステップS610）。次いで、素イデアル1（秘密鍵1）pを生成して、データ番号4に格納し（ステップS620）、次いで、素イデアル2（秘密鍵1）qを生成して、データ番号5に格納する（ステップS640）。次いで、素イデアルp，qからその積である公開鍵1，n=p·qを求めて、データ番号6に格納する（ステップS661）。

【0140】次いで、p2-1とq2-1との最小公倍数であるLを計算し、これをデータ番号7に格納する（ステップS663）。次いで、e·d≡1.（mod L）となる2数、eおよびdを求め、秘密鍵2dをデータ番号8に、公開鍵2eをデータ番号9に、それぞれ格納して（ステップS665）、鍵生成を終了する。

【0141】図14は、判別式Dの値を格納するルーチンの詳細を示すフローチャートであり、ステップS610以下の詳細を示す。まず、データ番号1から有理整数mを読み出し、m≡1（mod 4）か否かを判定する（ステップS612）。m≡1（mod 4）であれば、判別式D=mとし（ステップS614）、m≡1（mod 4）でなければ（m≡2，3（mod 4））、判別式D=4mとして（ステップS616）、判別式Dの値をデータ番号2に格納し（ステップS618）、次の処理へ移る。

【0142】図15は、2次体における素イデアル1，pの生成処理の詳細を示すフローチャートであり、ステップS620以下の詳細を示す。まず、所定の範囲の有理整数pを任意に（ランダムに）生成し、データ番号3に格納する（ステップS622）。次いで、ルジャンドルの記号（D／p）2を計算する（ステップS624）。（D／p）2の値は、例えばユークリッドの互除法を用いて計算することができる。

【0143】次いで、この値が-1であるか否かを判定する（ステップS626）。この判定で、（D／p）2=-1でなければ、ステップS622に戻り、再度有理素数pを生成する。ステップS626の判定で、（D／p）2=-1であれば、pは素イデアルであると判定できるので、素イデアル1（秘密鍵1）pをデータ番号4に格納して（ステップS628）、次の処理へ移る。

【0144】図16は、2次体における素イデアル2，qの生成処理の詳細を示すフローチャートであり、ステップS640以下の詳細を示す。まず、所定の範囲の有理整数qを任意に（ランダムに）生成する（ステップS642）。次いで、qがpに等しいか否かを判定し（ステップS644）、等しければpと異なるqを生成するために、ステップS642に戻る。等しくなければ、次いで、ルジャンドルの記号（D／p）2を計算し（ステップS646）、この値が-1であるか否かを判定する（ステップS648）。

【0145】ステップS648の判定で、（D／p）2=-1でなければ、ステップS642に戻り、再度有理素数qを生成する。ステップS648の判定で、（D／p）2=-1であれば、qは素イデアルであると判定できるので、素イデアル2（秘密鍵1）qをデータ番号5に格納して（ステップS650）、次の処理へ移る。

【0146】図17ないし図20は、暗号化送信処理の

詳細手順を示すフローチャートである。暗号化送信処理では、表３に示すようなメモリ上のデータ配置を行う。

【０１４７】

【表３】

データ番号	データ内容
1	公開鍵１（n）
2	公開鍵２（e）
3	ブロック数（r）
4	分割平文長（log2nを超えない最大整数）
5	平文M
6	平文ブロックM1
…	…
5+r	平文ブロックMr
6+r	暗号文ブロックC1
…	…
5+2r	暗号文ブロックCr

そして、図１７に示すように、まずステップＳ７０１で、暗号化の為の初期設定が行われる。すなわちステップＳ７０１では、公開鍵１（ｎ）、公開鍵１ｅ、ブロック数ｒ、分割平文長（ｌｏｇ２ｎを超えない最大整数）をそれぞれデータ番号１～４に設定する。

【０１４８】次いで、平文Ｍをデータ番号５に読み込む（ステップＳ７０３）。次いで、平文Ｍの先頭からそれぞれ分割平文長のビット数の長さのブロックをｒ個切り出す平文分割処理を行い（ステップＳ７１０）、このｒ個のブロック毎に暗号化を行う（ステップＳ７３０）。暗号化されたブロックは、暗号送信処理され（ステップＳ７５０）、また平文が残っていれば、平文分割処理（ステップＳ７１０）に戻り、平文が残っていなければ終了する。

【０１４９】図１８は、平文分割処理の詳細を示すフローチャートであり、ステップＳ７１０以下の詳細を示す。まず、平文分割処理の制御変数ｉ，ｊ，ｋの初期設定を行う（ステップＳ７１２）。制御変数ｉは、分割された平文ブロックの番号を示すものである。制御変数ｊは、元の平文データＭを構成する各ビットのビット番号である。ｋは、ビット数で表した分割平文長である。ステップＳ７１２では、それぞれ、ｉ＝０，ｊ＝０，ｋ＝（ｌｏｇ２ｎを超えない最大整数）と設定する。

【０１５０】次いで、データ番号５からＭを読み出し、Ｍのｊビット目から（ｊ＋ｋ－１）ビット目までを切り出し、これをＭｉとしてデータ番号（６＋ｉ）に格納する（ステップＳ７１４）。次いで、制御変数更新のために、ｉ＝ｉ＋１，ｊ＝ｊ＋ｋとする（ステップＳ７１６）。次いで、所定のブロック数ｒまで分割されたか否かを判定するため、ｉとｒとを比較する（ステップＳ７１８）。ステップＳ７１８の判定において、ｉ≠ｒであれば、まだ分割すべきブロックが残っているので、ステップＳ７１４へ戻る。

【０１５１】ステップＳ７１８の判定において、ｉ＝ｒであれば、分割すべきブロックが残っていないので、ｊがＭの最終ビット番号未満か否かを判定する（ステップＳ７２０）。ｊがＭの最終ビット番号未満であれば、Ｍのｊビット目から最後までをＭｉとして、データ番号５に格納して（ステップＳ７２２）次の処理へ移る。ｊがＭの最終ビット番号以上であれば、ＡＬＬ‘０’をデータ番号５に格納して（ステップＳ７２４）次の処理へ移る。

【０１５２】図１９は、暗号化処理の詳細を示すフローチャートであり、ステップＳ７３０以下の詳細を示す。まず、暗号化処理の制御変数ｉの初期設定、ｉ＝０を行う（ステップＳ７３２）。制御変数ｉは、分割された平文ブロックの番号を示すものである。次いで、データ番号（６＋ｉ）から平文ブロックＭｉを読み出し、これにイデアル（ｎ）を法とするｅ乗演算を行い、その結果を暗号化ブロックＣｉとして、データ番号（６＋ｒ＋ｉ）に格納する（ステップＳ７３４）。次いで、制御変数ｉを１だけ増加させる（ステップＳ７３６）。

【０１５３】次いで、所定のブロック数ｒまで暗号化されたか否かを判定するため、ｉとｒとを比較する（ステップＳ７３８）。ステップＳ７３８の判定において、ｉ≠ｒであれば、また暗号化すべきメッセージブロックが残っているので、ステップＳ７３４へ戻る。ステップＳ７３８の判定において、ｉ＝ｒであれば、暗号化が終了したので、次の処理へ移る。

【０１５４】図２０は、暗号文送信処理の詳細を示すフローチャートであり、ステップＳ７５０以下の詳細を示す。まず、暗号文送信処理の制御変数ｉの初期設定、ｉ＝０を行う（ステップＳ７５２）。制御変数ｉは、暗号化ブロックの番号を示すものである。次いで、データ番号（６＋ｒ＋ｉ）から暗号化ブロックＣｉを読み出し、送信する（ステップＳ７５４）。次いで、制御変数ｉを１だけ増加させる（ステップＳ７５６）。

【０１５５】次いで、所定のブロック数ｒまで送信されたか否かを判定するため、ｉとｒとを比較する（ステップＳ７５８）。ステップＳ７５８の判定において、ｉ≠ｒであれば、また送信すべき暗号化ブロックが残っているので、ステップＳ７５４へ戻る。ステップＳ７５８の判定において、ｉ＝ｒであれば、送信が終了したので、データ番号５のＭｉを読み出し、Ｍｉ＝ＡＬＬ‘０’か否かを判定する（ステップＳ７６０）。

【０１５６】ステップＳ７６０の判定において、ＡＬＬ‘０’でなければ、まだ分割以下の処理を行うべき平文が残っているので、平文分割処理（ステップＳ７１０）へ戻る。ステップＳ７６０の判定において、ＡＬＬ‘０’であれば、暗号化送信処理が終了する。

【０１５７】図２１ないし図２３は、受信復号化処理の詳細手順を示すフローチャートである。

【０１５８】受信復号化処理では、表４に示すようなメモリ上のデータ配置を行う。

【０１５９】

【表４】

データ番号	データ内容
1	公開鍵１（n）
2	秘密鍵２　d
3	ブロック数　r
4	分割平文長［log2n］
5	暗号文ブロックC0
…	…
4＋r	暗号文ブロックCr-1
4＋r＋1	平文ブロックM0
…	…
4＋2r	平文ブロックMr-1
4＋2r＋1	結合平文M

そして、図２１に示すように、まずステップＳ８０１で、復号化の為の初期設定が行われる。すなわちステップＳ８０１では、公開鍵１（n）、秘密鍵２ｄ、ブロック数ｒ、分割平文長（log2nを超えない最大整数）をそれぞれデータ番号１～４に設定する。次いで、暗号文ブロックC0～Cr-1をブロック毎にデータ番号５～（５＋ｒ－１）へブロック毎に読み込む（ステップＳ８０３）。次いで、暗号文ブロック毎の復号化処理（ステップＳ８１０）、続いて平文統合処理（ステップＳ８２０）を行う。

【０１６０】図２２は、復号化処理の詳細を示すフローチャートであり、ステップＳ８１０以下の詳細を示す。まず、復号化処理の制御変数ｉの初期設定、ｉ＝０を行う（ステップＳ８１２）。制御変数ｉは、暗号化ブロックの番号を示すものである。次いで、データ番号（５＋ｉ）から暗号化ブロックCiを読み出し、これにイデアル（n）を法とするｄ乗演算を行い、その結果を復号化された平文ブロックMiとして、データ番号（５＋ｒ＋ｉ）に格納する（ステップＳ８１４）。次いで、制御変数ｉを１だけ増加させる（ステップＳ８１６）。

【０１６１】次いで、所定のブロック数ｒまで復号化されたか否かを判定するため、ｉとｒとを比較する（ステップＳ８１８）。ステップＳ８１８の判定において、ｉ≠ｒであれば、また復号化すべきブロックが残っているので、ステップＳ８１４へ戻る。ステップＳ８１８の判定において、ｉ＝ｒであれば、復号化が終了したので、次の平文統合処理へ移る。

【０１６２】図２３は、平文統合処理の詳細を示すフローチャートであり、ステップＳ８２０以下の詳細を示す。まず、平文統合処理の制御変数ｉ、ｊ、ｋの初期設定を行う（ステップＳ８２２）。制御変数ｉは、統合すべき平文ブロックの番号を示すものである。制御変数ｊは、統合後の平文データMを構成するｉビットのビット番号である。ｋは、ビット数で表した分割平文長である。ステップＳ８２２では、それぞれ、ｉ＝０，ｊ＝０，ｋ＝（log2nを超えない最大整数）と設定する。

【０１６３】次いで、データ番号（５＋ｒ＋ｉ）からMiを読み出し、作業領域のｊビット目から（ｊ＋ｋ－１）ビット目までに格納する（ステップＳ８２４）。次いで、制御変数更新のために、ｉ＝ｉ＋１，ｊ＝ｊ＋ｋとする（ステップＳ８２６）。次いで、所定のブロック数ｒまで統合されたか否かを判定するため、ｉとｒとを比較する（ステップＳ８２８）。ステップＳ８２８の判定において、ｉ≠ｒであれば、また統合すべきブロックが残っているので、ステップＳ８２４へ戻る。

【０１６４】ステップＳ８２８の判定において、ｉ＝ｒであれば、統合すべきブロックが残っていないので、作業領域内に連接されたｒ個のブロックをデータ番号（５＋２ｒ）の末尾に連接して格納して（ステップＳ８３０）、受信復号化処理を終了する。

【０１６５】［３次円分体（アイゼンシュタイン体）を使用した公開鍵暗号方式の数値例］次に、３次円分体（アイゼンシュタイン体）を使用した本発明に係る公開鍵暗号方式の実施の形態を具体的な数値例を用いて説明する。

【０１６６】［鍵生成処理］２個の素イデアルP＝（２９），Q＝（４１）（秘密鍵１）を生成し、その積N＝（１１８９）の剰余類（公開鍵１）を決定する。次に秘密鍵１からL＝１６８０を計算し、e＝１２７（公開鍵２）と、d＝４６３（秘密鍵２）を生成する。

【０１６７】［平文分割処理］平文M＝１２３４５６に対して、剰余類の範囲内の分割平文（M0，M1）＝（１２３，４５６）を生成する。

【０１６８】［暗号化処理］受け手の公開鍵e＝１２７を用い、

【数４４】

$$(C0, C1) \equiv (123, 456)^{127} \equiv (164, 1004) \bmod (1189) \cdots (46)$$

式（４６）によって暗号化を行ない、受け手に送る。

【０１６９】［復号化処理］受け手は自らの秘密鍵d＝４６３を用い、

【数４５】

$$(164, 1004)^{463} \equiv (123, 456) \equiv (M0, M1) \bmod (1189) \cdots (47)$$

式（４７）を計算することによって、復号化を行なう。

【０１７０】［平文統合処理］復号化された分割平文

（M0，M1）＝（123，456）を連接し、原平文M＝123456へ復元する。

【0171】［2次体（ガウス体）を利用した公開鍵暗号方式の数値例］次に、2次体（ガウス体）を利用した本発明に係る公開鍵暗号方式の実施形態を具体的な数値例を用いて説明する。

【0172】［鍵生成処理］秘密鍵1である2個の素イデアルP＝（31），Q＝（47）を生成し、その積N＝（1457）の剰余類（公開鍵1）を決定する。

【0173】次いで、秘密鍵1からL＝22080を計算し、e＝127（公開鍵2）と、d＝20863（秘密鍵2）を生成する。

【0174】［平文分割処理］平文M＝123456に対して、それぞれ剰余類の範囲内の大きさのブロックに分割した分割平文（M0，M1）＝（123，456）を生成する。

【0175】［暗号化処理］受け手の公開鍵e＝127を用いて、

【数46】

$$(C0,\ C1) \equiv (123,\ 456)^{127} \equiv (246,\ 545) \bmod (1457) \quad \cdots (48)$$

式（48）によって暗号化を行い、暗号文（C0，C1）＝（246，545）を受け手に送る。

【0176】［復号化処理］受け手は自らの秘密鍵d＝20836を用いて、

【数47】

$$(246,\ 545)^{20836} \equiv (M0,\ M1) \bmod (1457) \quad \cdots (49)$$

式（49）を計算することによって、暗号文の復号化を行い、分割された平文（M0，M1）＝（123，456）を得る。

【0177】［平文統合処理］復号化された分割平文（M0，M1）＝（123，456）を連接し、原平文M＝123456へ復元する。

【0178】次に、本発明に係る公開鍵暗号を利用した認証方式及び認証装置について説明する。図2は、本発明に係る公開鍵暗号による認証通信装置111の全体構成を示すブロック図である。

【0179】本発明の公開鍵暗号方式を用い、認証を行ないたい者が、自らの秘密鍵により認証文の暗号化を行なって生成した認証子を受け手に送る方式による認証装置を構成することができる。

【0180】図2によれば、認証通信装置111は、鍵生成装置21と、認証文生成装置131と、認証文検証装置141と、通信路51とを含んで構成される。鍵生成装置21と、通信路51とは、図1の暗号化通信装置11に使用したものと同じであるのでその説明は簡略にする。

【0181】［認証用鍵生成処理］鍵生成装置21は、2個の素イデアルP，Q（秘密鍵1）を生成し、その積N＝PQの剰余類（公開鍵1）を決定する。次に素イデアルP，QからLを計算し、e（公開鍵2）と、d（秘密鍵2）を生成する。

【0182】次いで、2個の素イデアル（秘密鍵1）の生成および剰余類（公開鍵1）の決定を行う。円分体の場合は、原始根の位数mを入力として、秘密鍵1である素イデアルP，Qを出力する。2次体の場合は、判別式Dを入力として、秘密鍵1である素イデアルP，Qを出力し、e＝127（公開鍵2）と、d＝20863（秘密鍵2）を生成する。

【0183】次いで、公開鍵2であるeと秘密鍵2であるdの生成を行う。円分体、2次体ともに、秘密鍵1であるP，Qを入力として、公開鍵2であるeと秘密鍵2であるdを出力する。また、鍵生成装置21は、ブロック数r、及び分割平文長［log2 n］を認証文生成装置131へ出力する。以上の鍵生成処理は、図1の暗号化通信装置11における鍵生成装置21の動作と同じである。

【0184】［認証文生成処理］認証文生成装置131は、認証文を受け入れる認証文入力部133と、認証文をハッシュ化して認証子を生成する認証文ハッシュ化処理部135と、認証子をブロックに分割する認証子分割処理部137と、分割認証子を暗号化する認証子暗号化処理部139とを含んで構成されている。

【0185】認証文入力部133により受け入れられた認証文Mは、認証文ハッシュ化処理部135により、ハッシュ関数hを用いてハッシュ化され、認証子h（M）が生成される。ハッシュ関数hは特に限定されるものではない。

【0186】次いで、認証子h（M）は、認証子分割処理部137により、それぞれ分割平文長［log2 n］の長さのr個のブロックずつに分割される。ここで、rをZ上の整数環Oの次元とする。認証子h（M）に対して、剰余類の範囲内の分割認証子h（M）＝（h（M）0，h（M）1，…，h（M）r-1）を生成する。

【0187】次いで、認証子暗号化処理部139により、送り手の秘密鍵dを用いて、

【数48】

$$(h(C)0,\ h(C)1,\ \cdots,\ h(C)r\text{-}1) \equiv (h(M)0,\ h(M)1,\ \cdots,\ h(M)r\text{-}1)^{d} \bmod N \quad \cdots (50)$$

式（50）によって認証子の暗号化を行なう。

【0188】以上より得られた暗号化認証子h（C）＝（h（C）0，h（C）1，…，h（C）r-1）と認証文Mの組を通信路51を介して、受け手すなわち認証

文検証装置１４１に送る。

【０１８９】認証文検証装置１４１は、認証子復号化処理部１４３と、認証子統合処理部１４５と、認証子ハッシュ化処理部１４７と、認証子照合部１４９とを含んで構成されている。

【０１９０】認証文検証装置１４１は、認証文生成装置１３１から暗号化認証子ｈ（Ｃ）＝（ｈ（Ｃ）０，ｈ（Ｃ）１，…，ｈ（Ｃ）ｒ－１）

$$= (h(C)0, h(C)1, \cdots, h(C)r-1)^e \equiv (h(M)0, h(M)1, \cdots, h(M)r-1) \pmod{N} \quad \cdots (S1)$$

式（Ｓ１）を計算することによって、暗号化認証子を復号化して分割認証子を得る。

【０１９２】次いで、認証子統合処理部１４５は、復号化された分割認証子（ｈ（Ｍ）０，ｈ（Ｍ）１，…，ｈ（Ｍ）ｒ－１）を統合し、認証子ｈ（Ｍ）を生成する。一方、認証文ハッシュ化処理部１４７は、送られて来た認証文Ｍをハッシュ関数ｈにより、ハッシュ化して認証子ｈ（Ｍ）′とする。

【０１９３】次いで、認証子照合部１４９により、認証子ｈ（Ｍ）とｈ（Ｍ）′とを比較し、一致していれば認証文が正当であり、不一致であれば認証文が不当であるとして、認証文の正当性を判断する。

【０１９４】図２４ないし図２７は、認証文生成処理の詳細手順を示すフローチャートである。認証文生成処理では、表５に示すようなメモリ上のデータ配置を行う。

【０１９５】

【表５】

データ番号	データ内容
１	公開鍵１（ｎ）
２	秘密鍵２（ｄ）
３	ブロック数ｒ
４	分割平文長［ｌｏｇ２ｎ］
５	認証文Ｍ
６	認証子ｈ（Ｍ）
ｉ	認証子ブロックｈ（Ｍ）ｉ
…	…
６＋ｒ－１	認証子ブロックｈ（Ｍ）ｒ－１
６＋ｒ	暗号化認証子ブロックｈ（Ｃ）０
…	…
６＋２ｒ－１	暗号化認証子ブロックｈ（Ｃ）ｒ－１

そして、図２４に示すように、まずステップＳ９０１で、認証文生成の為の初期設定が行われる。すなわちステップＳ９０１では、公開鍵１（ｎ）、秘密鍵２ｄ、ブロック数ｒ、分割平文長（ｌｏｇ２ｎを超えない最大整数）をそれぞれデータ番号１～４に設定する。次いで、認証文Ｍをデータ番号５に読み込む（ステップＳ９０３）。

【０１９６】次いで、認証文Ｍをデータ番号５から読み出し、ハッシュ関数ｈでハッシュ化し、その結果である認証子ｈ（Ｍ）をデータ番号６に格納する（ステップＳ９０５）。

【０１９７】次いで、認証子ｈ（Ｍ）の先頭からそれぞれ分割平文長のビット数の長さのブロックをｒ個切り出す認証子分割処理を行い（ステップＳ９２０）、続いで、認証子暗号化処理を行う（ステップＳ９４０）。暗号化された認証子ブロックは、暗号送信処理され（ステップＳ９６０）、また認証子が残っていれば、認証子分割処理（ステップＳ９２０）に戻り、認証子が残っていなければ終了する。

【０１９８】図２５は、認証子分割処理の詳細を示すフローチャートであり、ステップＳ９２０以下の詳細を示す。まず、認証子分割処理の制御変数ｉ，ｊ，ｋの初期設定を行う（ステップＳ９２２）。制御変数ｉは、分割された認証子ブロックｈ（Ｍ）ｉの番号ｉを示すものである。制御変数ｊは、分割前の認証子ｈ（Ｍ）を構成する各ビットのビット番号である。ｋは、ビット数で表した分割平文長である。ステップＳ９２２では、それぞれ、ｉ＝０，ｊ＝０，ｋ＝（ｌｏｇ２ｎを超えない最大整数）と設定する。

【０１９９】次いで、データ番号６から認証子ｈ（Ｍ）

を読み出し、h（M）のjビット目から（j＋k－1）ビット目までを切り出し、これをh（M）iとしてデータ番号（7＋i）に格納する（ステップＳ９２４）。次いで、制御変数更新のために、i＝i＋1，j＝j＋kとする（ステップＳ９２６）。次いで、所定のブロック数rまで分割されたか否かを判定するため、iとrとを比較する（ステップＳ９２８）。ステップＳ９２８の判定において、i≠rであれば、また分割すべきブロックが残っているので、ステップＳ９２４へ戻る。

【０２００】ステップＳ９２８の判定において、i＝rであれば、分割すべきブロックが残っていないので、jがh（M）の最終ビット番号未満か否かを判定する（ステップＳ９３０）。jがh（M）の最終ビット番号未満であれば、h（M）のjビット目から最後までをh（M）′として、データ番号５に格納して（ステップＳ９３２）、次の認証子暗号化処理へ移る。jがh（M）の最終ビット番号以上であれば、ＡＩＩ‘０’をデータ番号５に格納して（ステップＳ９３４）、次の認証子暗号化処理へ移る。

【０２０１】図２６は、認証子暗号化処理の詳細を示すフローチャートであり、ステップＳ９４０以下の詳細を示す。まず、認証子暗号化処理の制御変数iの初期設定、i＝０を行う（ステップＳ９４２）。制御変数iは、分割された認証子ブロックの番号を示すものである。次いで、データ番号（7＋i）から認証子ブロックh（M）iを読み出し、これにイデアル（n）を法とするd乗演算を行い、その結果を暗号化認証子ブロックh（C）iとして、データ番号（7＋r＋i）に格納する（ステップＳ９４４）。次いで、制御変数iを１だけ増加させる（ステップＳ９４６）。

【０２０２】次いで、所定のブロック数rまで暗号化されたか否かを判定するため、iとrとを比較する（ステップＳ９４８）。ステップＳ９４８の判定において、i≠rであれば、また暗号化すべき認証子ブロックが残っているので、ステップＳ９４４へ戻る。ステップＳ９４８の判定において、i＝rであれば、暗号化が終了したので、次の暗号文送信処理へ移る。

【０２０３】図２７は、暗号文送信処理の詳細を示すフローチャートであり、ステップＳ９６０以下の詳細を示す。まず、暗号文送信処理の制御変数iの初期設定、i＝０を行う（ステップＳ９６２）。制御変数iは、暗号化ブロックの番号を示すものである。次いで、データ番号（7＋r＋i）から暗号化認証子ブロックh（C）iを読み出し、送信する（ステップＳ９６４）。次いで、制御変数iを１だけ増加させる（ステップＳ９６６）。

【０２０４】次いで、所定のブロック数rまで送信されたか否かを判定するため、iとrとを比較する（ステップＳ９６８）。ステップＳ９６８の判定において、i≠rであれば、また送信すべき暗号化ブロックが残っているので、ステップＳ９６４へ戻る。ステップＳ９６８の判定において、i＝rであれば、送信が終了したので、データ番号６のh（M）′を読み出し、h（M）′＝ＡＩＩ‘０’か否かを判定する（ステップＳ９７０）。

【０２０５】ステップＳ９７０の判定において、ＡＩＩ‘０’でなければ、また分割以下の処理を行うべき認証子が残っているので、認証子分割処理（ステップＳ９２０）へ戻る。ステップＳ９７０の判定において、ＡＩＩ‘０’であれば、認証文生成処理が終了する。

【０２０６】図２８ないし図３１は、認証文復号化処理の詳細手順を示すフローチャートである。

【０２０７】認証文復号化処理では、表６に示すようなメモリ上のデータ配置を行う。

【０２０８】

【表６】

データ番号	データ内容
1	公開鍵 e（1）
2	公開鍵 e（2）
3	ブロック数 r
4	分割平文長 L［log n］
5	認証文M
6	暗号化認証子ブロック h（C）1
…	…
6＋i	暗号化認証子ブロック h（C）i
6＋r＋1	認証子ブロック h（M）1
…	…
6＋2r	認証子ブロック h（M）r
6＋2r＋1	結合認証子 h（M）
6＋2r＋2	ハッシュ化認証文 h（M）′

そして、認証文復号化処理では、まず図２８に示すように、ステップＳ１００１で、復号化の為の初期設定が行

われる。すなわちステップＳ１００１では、公開鍵１（ｎ）、公開鍵２ｅ、ブロック数ｒ、分割平文長（ｌｏｇ２ｎを超えない最大整数）をそれぞれデータ番号１～４に設定する。

【０２０９】次いで、認証文Ｍをデータ番号５へ、暗号化認証子ブロックｈ（Ｃ）０～ｈ（Ｃ）ｒ－１をブロック毎にデータ番号６～（５＋ｒ）へブロック毎に読み込む（ステップＳ１００３）。次いで、ブロック毎の認証子復号化処理（ステップＳ１０２０）、続いて認証子統合処理（ステップＳ１０４０）、認証文ハッシュ化処理（ステップＳ１０６０）、認証確認処理（ステップＳ１０８０）を順次行う。

【０２１０】図２９は、認証子復号化処理の詳細を示すフローチャートであり、ステップＳ１０２０以下の詳細を示す。まず、認証子復号化処理の制御変数ｉの初期設定、ｉ＝０を行う（ステップＳ１０２２）。制御変数ｉは、暗号化認証子ブロックの番号を示すものである。次いで、データ番号（６＋ｉ）から暗号化認証子ブロックｈ（Ｃ）ｉを読み出し、これにイデアル（ｎ）を法とするｅ乗演算を行い、その結果を復号化された認証子ブロックｈ（Ｍ）ｉとして、データ番号（６＋ｒ＋ｉ）に格納する（ステップＳ１０２４）。次いで、制御変数ｉを１だけ増加させる（ステップＳ１０２６）。

【０２１１】次いで、所定のブロック数ｒまで復号化されたか否かを判定するため、ｉとｒとを比較する（ステップＳ１０２８）。ステップＳ１０２８の判定においてｉ≠ｒであれば、まだ復号化すべきブロックが残っているので、ステップＳ１０２４へ戻る。ステップＳ１０２８の判定において、ｉ＝ｒであれば、復号化が終了したので、次の認証子統合処理へ移る。

【０２１２】図３０は、認証子統合処理の詳細を示すフローチャートであり、ステップＳ１０４０以下の詳細を示す。まず、認証子統合処理の制御変数ｉ、ｊ、ｋの初期設定を行う（ステップＳ１１０４２）。制御変数ｉは、統合すべき認証子ブロックｈ（Ｍ）ｉの番号ｉを示すものである。制御変数ｊは、統合後の認証子データｈ（Ｍ）を構成する各ビットのビット番号である。ｋは、ビット数で表した分割平文長である。ステップＳ１０４２では、それぞれ、ｉ＝０、ｊ＝０、ｋ＝（ｌｏｇ２ｎを超えない最大整数）と設定する。

【０２１３】次いで、データ番号（６＋ｒ＋ｉ）からｈ（Ｍ）ｉを読み出し、作業領域のｊビット目から（ｊ＋ｋ－１）ビット目までに格納する（ステップＳ１０４４）。次いで、制御変数更新のために、ｉ＝ｉ＋１、ｊ＝ｊ＋ｋとする（ステップＳ１０４６）。

【０２１４】次いで、所定のブロック数ｒまで統合されたか否かを判定するため、ｉとｒとを比較する（ステップＳ１０４８）。ステップＳ１０４８の判定においてｉ≠ｒであれば、まだ統合すべきブロックが残っているので、ステップＳ１０４４へ戻る。

【０２１５】ステップＳ１０４８の判定において、ｉ＝ｒであれば、統合すべきブロックが残っていないので、作業領域内に連接して形成された統合認証子ｈ（Ｍ）をデータ番号（６＋２ｒ）の末尾に連接して格納し（ステップＳ１０５０）、次の認証文ハッシュ化処理へ移る。

【０２１６】図３１は、認証文ハッシュ化処理および認証確認処理の詳細を示すフローチャートであり、ステップＳ１０６０以下の詳細を示す。認証文ハッシュ化処理は、データ番号５から認証文Ｍを読み出し、ハッシュ関数ｈでハッシュ化し、その結果をｈ（Ｍ）＊として、データ番号（７＋２ｒ）に格納し（ステップＳ１０６２）、次の認証確認処理へ移る。

【０２１７】認証確認処理は、データ番号（６＋２ｒ）の統合認証子ｈ（Ｍ）とデータ番号（７＋２ｒ）のハッシュ化認証子ｈ（Ｍ）＊とを比較する（ステップＳ１０８２）。

【０２１８】この比較の結果、ｈ（Ｍ）とｈ（Ｍ）＊とが一致していれば、認証成功を出力して（ステップＳ１０８４）終了し、ｈ（Ｍ）とｈ（Ｍ）＊とが一致しなければ、認証失敗を出力して（ステップＳ１０８６）終了する。

【０２１９】なお、この認証通信の実施形態において、暗号化認証子ｈ（Ｃ）と認証文Ｍの組をさらに受信者の公開鍵を用いて暗号化する暗号化認証通信も、認証通信と暗号化通信とを組み合わせることによって従来の公開鍵暗号と同様に実現できる。

【０２２０】

【発明の効果】以上説明したように、本発明に係る公開鍵暗号方式によれば、従来のＲＳＡ暗号方式と比較して、完全読解についての安全性は同等以上の強度が得られるという効果を奏する。

【０２２１】また、本発明の公開鍵暗号方式に対して、従来の同報通信攻撃法は無効となるので、同報通信攻撃に対して著しく強度が増した暗号方式を提供することができるという効果を奏する。

【０２２２】また、本発明の公開鍵暗号方式によれば、暗号化にかかる計算時間の大部分を占める乗算回数が、楕円曲線上のＲＳＡ暗号と比較して２／５に削減され、大幅な高速化を実現することができるという効果を奏する。

【０２２３】また、本発明の拡大体への拡大次数が２次の場合は、従来のＲＳＡ暗号と比較して同程度の暗号化速度を確保しつつ、強度を高めることができるという効果を奏する。

【０２２４】また、本社から複数支社への同報データ通信や多地点間でのテレビ会議、有線あるいは無線電話などによる連絡などにおいて、守秘性を高めたいときに、本発明による暗号方式を用いてデータ、画像、音声の暗号化を行なうことにより、従来の方式に比べて同報通信攻撃に対して強度の増した通信を行なうことができると

いう効果を奏する。

【０２２５】また、本発明をインターネットなどのネットワークにおいて用いられているセキュリティ電子メイルなどの暗号通信に適用すれば、同報通信攻撃に対する強度を増すことができるという効果を奏する。

【０２２６】また、本発明に係る認証装置は、そのハードウェア資源ならびにソフトウェア資源のそれぞれ大部分を暗号装置と共用することができ、暗号化および認証の双方に対称使用が可能となるという効果を奏する。

【０２２７】また、本暗号装置および認証装置を使用することにより、同報通信攻撃に強いネットワーク管理プロトコルを設計することができるという効果を奏する。

【０２２８】また、ネットワークを利用して行なわれる電子商取引や電子金融取引等において、本暗号装置および認証装置を使用することにより、取引データの暗号化や取引相手の相互認証におけるセキュリティを向上させることができるという効果を奏する。

【０２２９】また、本発明の暗号化鍵生成方法および暗号化鍵生成装置によれば、代数体上の整数環における素イデアルを利用することにより、従来の有理整数環上の素数に比べて利用可能な暗号化鍵を増加させることができるという効果を奏する。

【図面の簡単な説明】

【図１】本発明に係る公開鍵暗号方式を用いた暗号化通信装置の概要を示すシステム構成図である。

【図２】本発明に係る公開鍵暗号方式による認証通信装置の概要を示すシステム構成図である。

【図３】円分体の素イデアル（秘密鍵１）生成処理部。

【図４】２次体の素イデアル（秘密鍵１）生成処理部。

【図５】円分体の公開鍵２，秘密鍵２の生成処理部。

【図６】平分分割処理部の概略フローチャートである。

【図７】イデアルｎを法とする乗算器の構成を示すブロック図である。

【図８】イデアルｎを法とする２乗演算器の構成を示すブロック図である。

【図９】円分体暗号における鍵生成処理のフローチャート図（１／４）であり、円分体暗号における鍵生成のメインルーチンを示す。

【図１０】円分体暗号における鍵生成処理のフローチャート図（２／４）であり、オイラー関数値φ（ｍ）の計算ルーチンを示す。

【図１１】円分体暗号における鍵生成処理のフローチャート図（３／４）であり、素イデアル１，ｐの生成ルーチンを示す。

【図１２】円分体暗号における鍵生成処理のフローチャート図（４／４）であり、素イデアル２，ｑの生成ルーチンを示す。

【図１３】２次体暗号における鍵生成処理のフローチャート図（１／４）であり、２次体暗号における鍵生成のメインルーチンを示す。

【図１４】２次体暗号における鍵生成処理のフローチャート図（２／４）であり、判別式Ｄの値を格納するルーチンを示す。

【図１５】２次体暗号における鍵生成処理のフローチャート図（３／４）であり、素イデアル１，ｐの生成ルーチンを示す。

【図１６】２次体暗号における鍵生成処理のフローチャート図（４／４）であり、素イデアル２，ｑの生成ルーチンを示す。

【図１７】暗号化送信処理のフローチャート図（１／４）であり、暗号化送信処理のメインルーチンを示す。

【図１８】暗号化送信処理のフローチャート図（２／４）であり、平文分割処理ルーチンを示す。

【図１９】暗号化送信処理のフローチャート図（３／４）であり、暗号化処理ルーチンを示す。

【図２０】暗号化送信処理のフローチャート図（４／４）であり、暗号文送信処理ルーチンを示す。

【図２１】受信復号化処理のフローチャート図（１／３）であり、受信復号化処理のメインルーチンを示す。

【図２２】受信復号化処理のフローチャート図（２／３）であり、復号化処理ルーチンを示す。

【図２３】受信復号化処理のフローチャート図（３／３）であり、平文統合処理ルーチンを示す。

【図２４】認証文生成処理のフローチャート図（１／４）であり、認証文生成処理のメインルーチンを示す。

【図２５】認証文生成処理のフローチャート図（２／４）であり、認証子分割処理ルーチンを示す。

【図２６】認証文生成処理のフローチャート図（３／４）であり、認証子暗号化ルーチンを示す。

【図２７】認証文生成処理のフローチャート図（４／４）であり、暗号文送信処理ルーチンを示す。

【図２８】認証文復号化処理のフローチャート図（１／４）であり、認証文復号化処理のメインルーチンを示す。

【図２９】認証文復号化処理のフローチャート図（２／４）であり、認証子復号化処理ルーチンを示す。

【図３０】認証文復号化処理のフローチャート図（３／４）であり、認証子統合処理ルーチンを示す。

【図３１】認証文復号化処理のフローチャート図（４／４）であり、認証子ハッシュ化処理ルーチンおよび認証確認処理ルーチンを示す。

【図３２】惰性する素数の積で生成されるイデアル（ｎ）による剰余類を２次元平面上に示した図である。

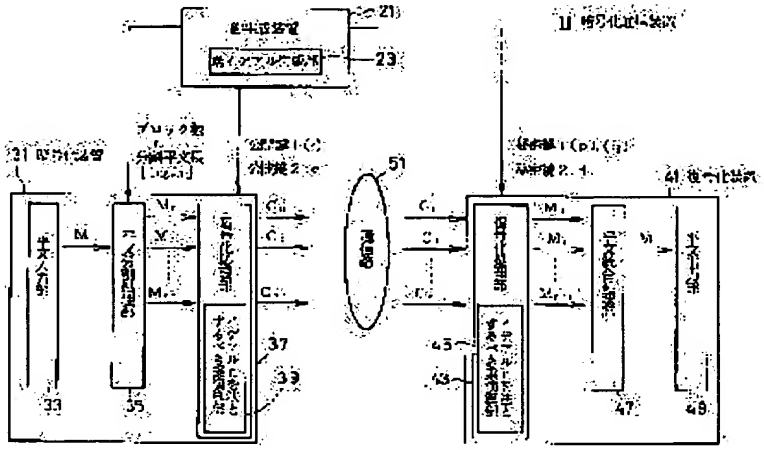
【図３３】従来のＲＳＡ暗号方式の概要を示すシステム構成図である。

【符号の説明】

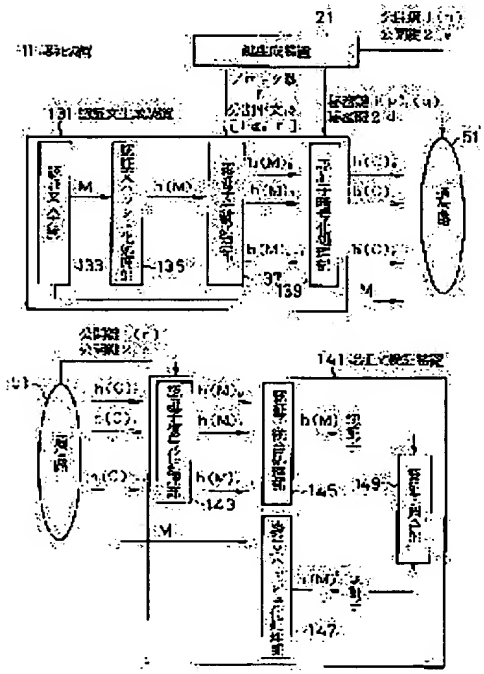
１１…暗号化通信装置、２１…鍵生成装置、２３…素イデアル生成部、３１…暗号化装置、３３…平文入力部、３５…平文分割部、３７…暗号化処理部、３９…イデアルｎを法とするべき乗演算部、４１…復号化装置、４３

…復号化処理部、４５…イデアルｎを法とするべき乗法算部、４７…平文統合処理部、４９…平文出力部、５１…通信路。

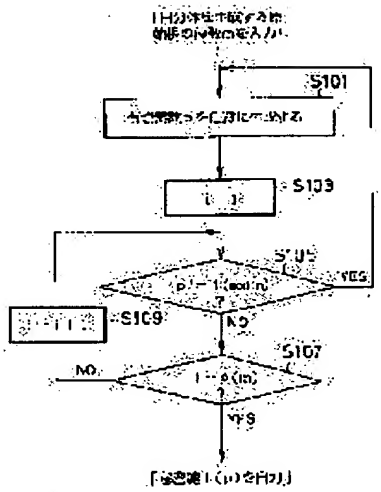
【図１】



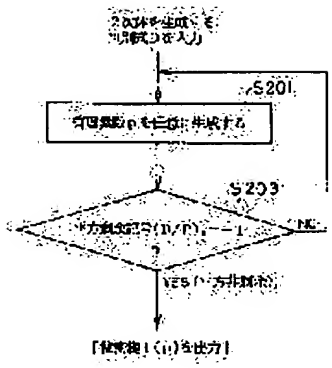
【図２】



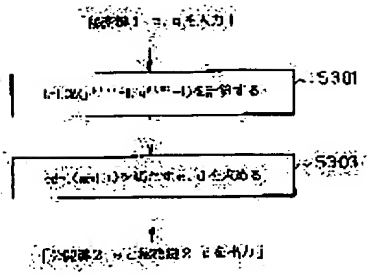
【図３】



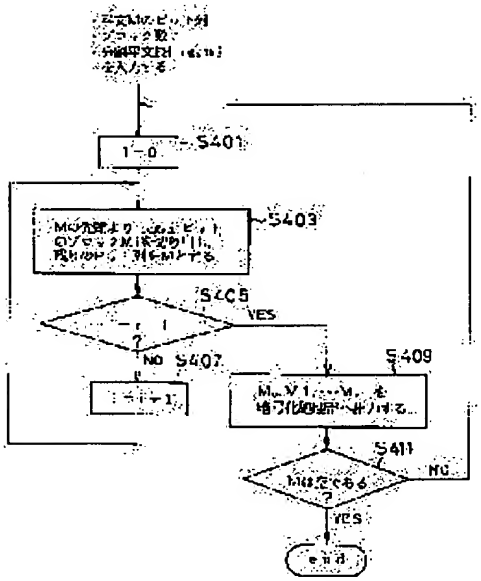
【図４】



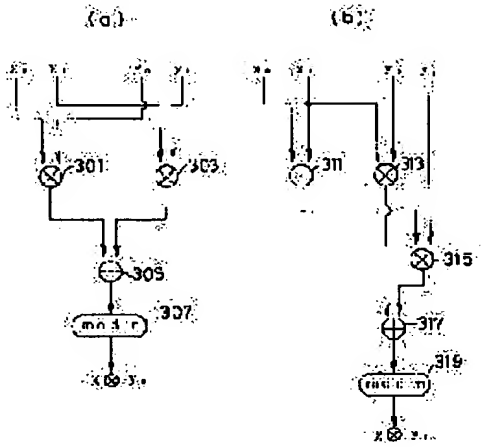
【図５】



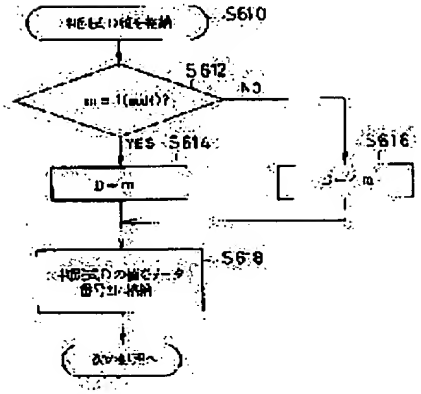
【図６】



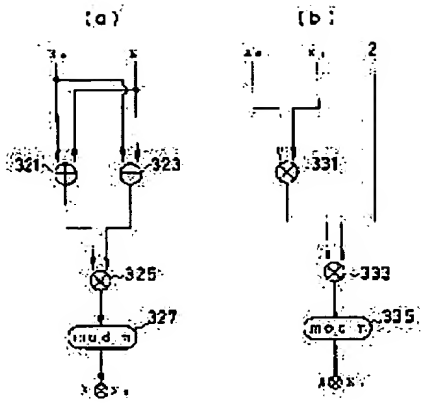
【図７】



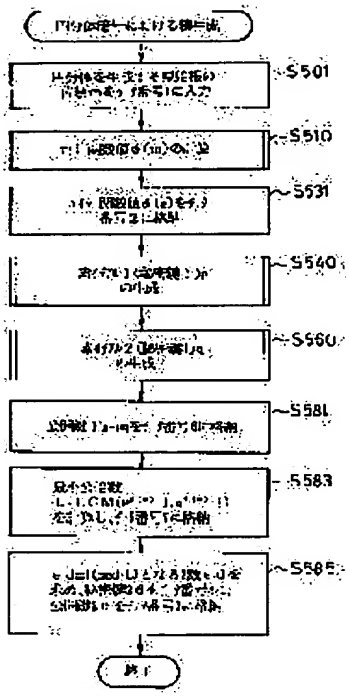
【図１４】



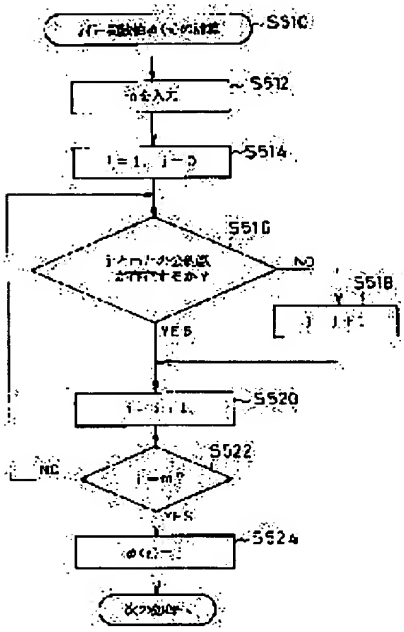
【図8】


(a)
(b)
321
323
325
327
331
333
335

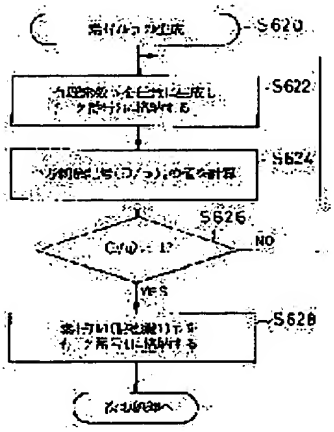
【図9】


S501
S510
S531
S540
S560
S581
S583
S585

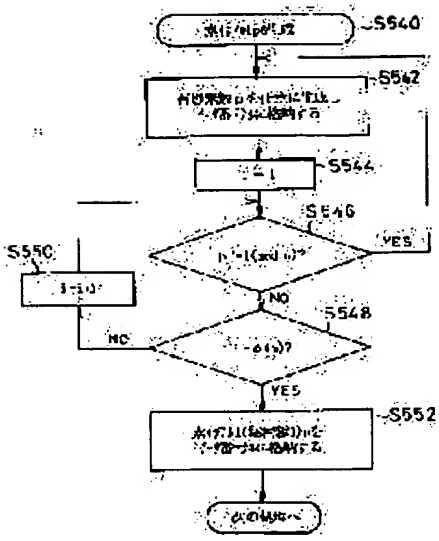
【図10】


S510
S512
S514
S516
NO
S518
YES
S520
S522
NO
YES
S524

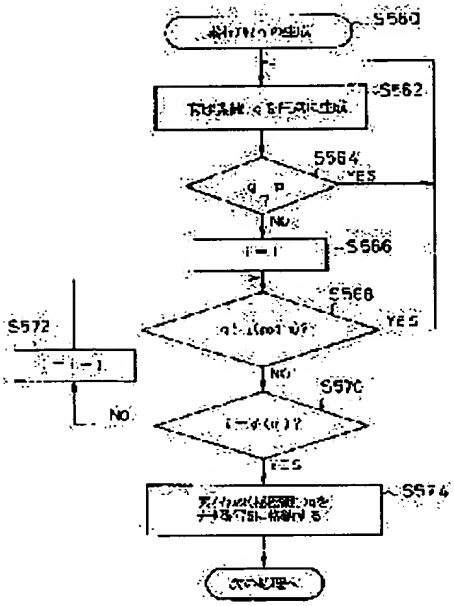
【図15】


S620
S622
S624
S626
NO
YES
S628

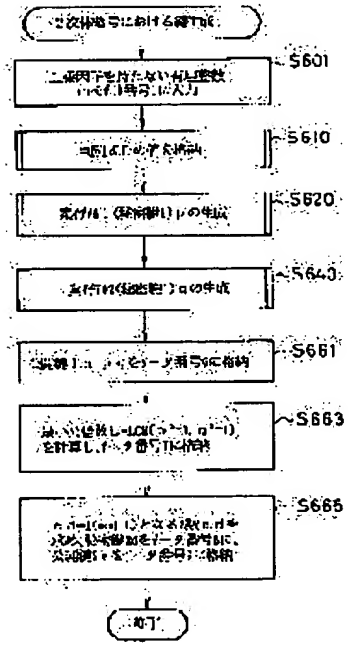
【図１１】



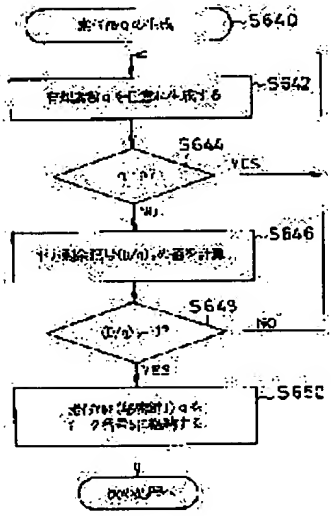
【図１２】



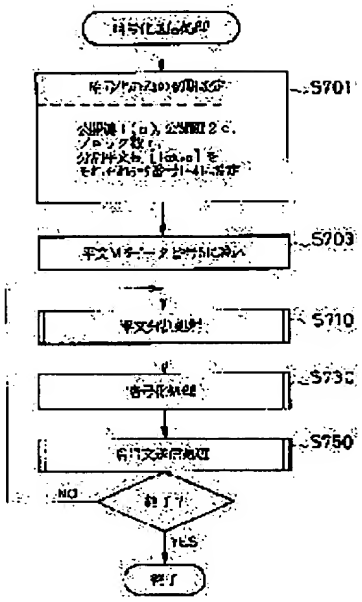
【図１３】



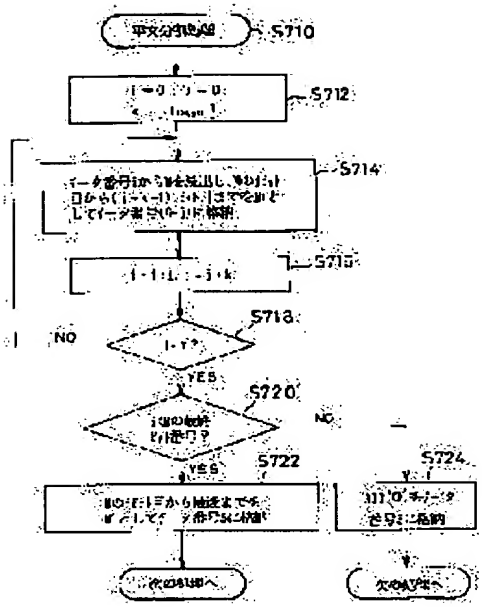
【図１６】



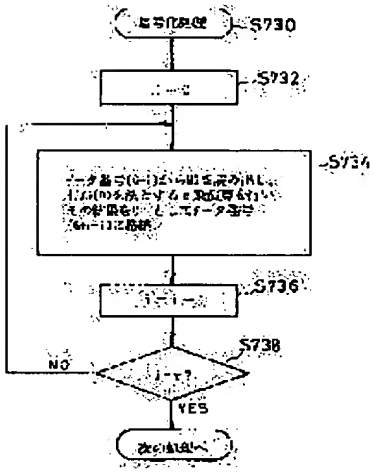
【図１７】



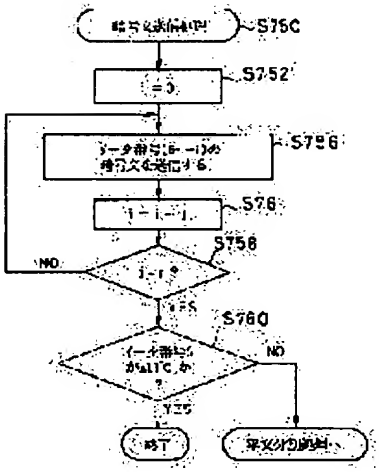
【図１８】



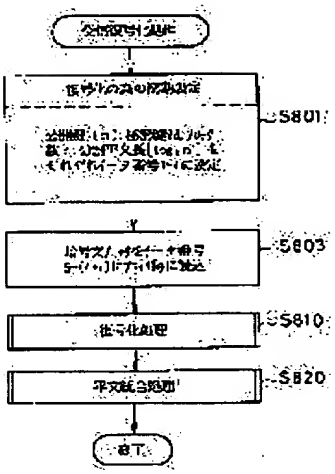
【図１９】



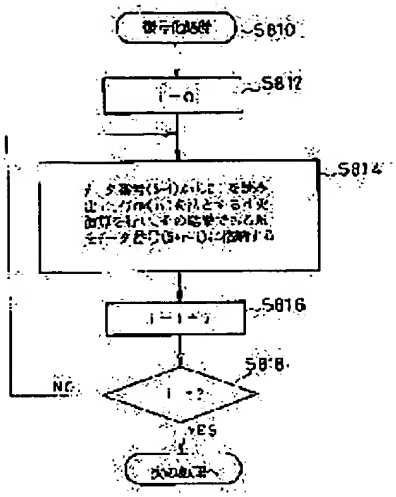
【図２０】



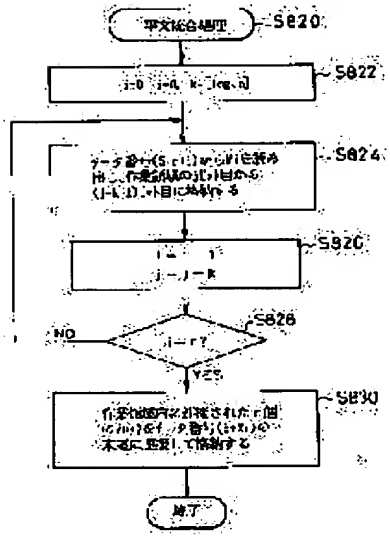
【図２１】



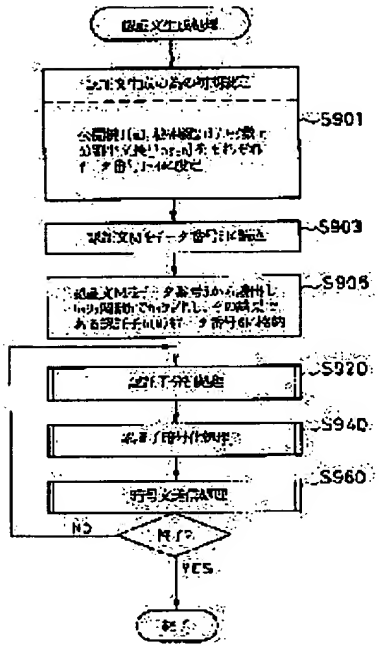
【図２２】



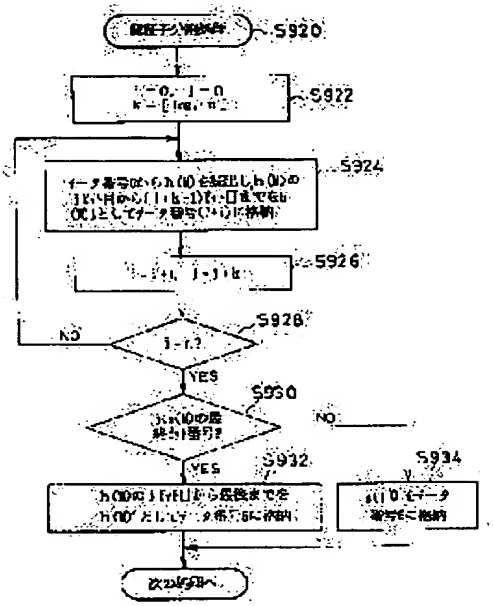
【図２３】



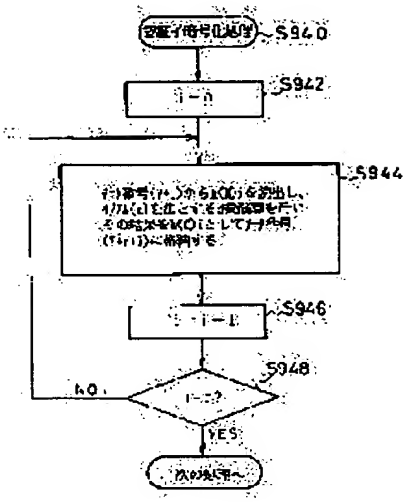
【図２４】



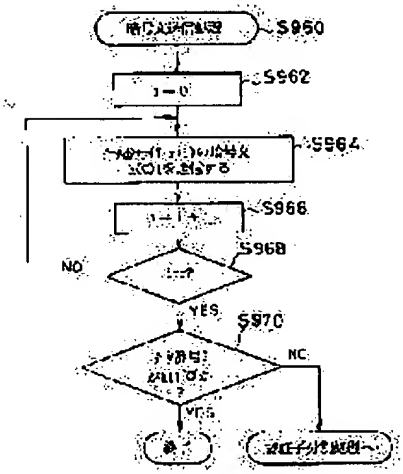
【図２５】



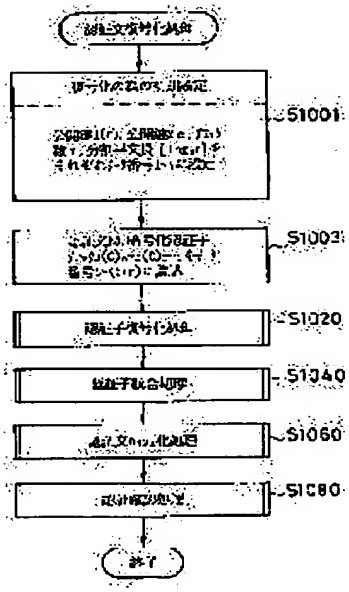
【図２６】



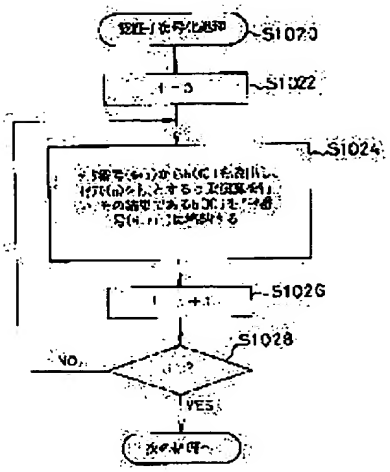
【図２７】



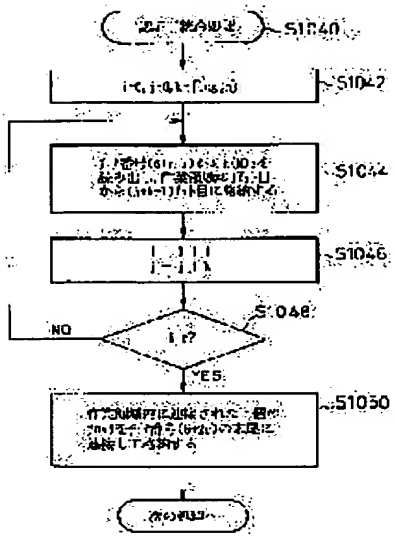
【図２８】



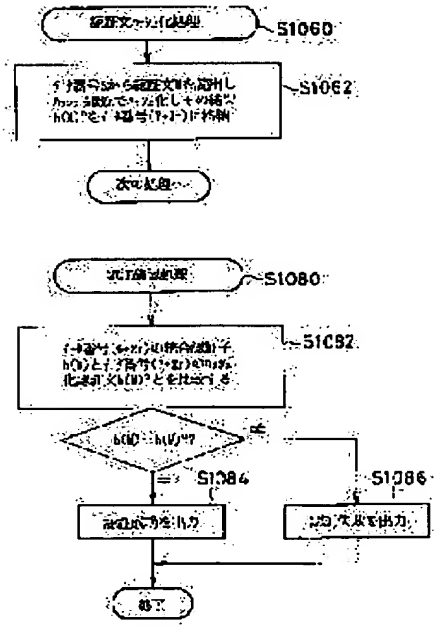
【図２９】



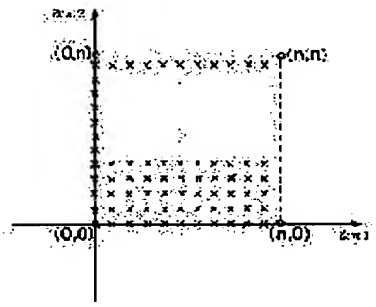
【図３０】



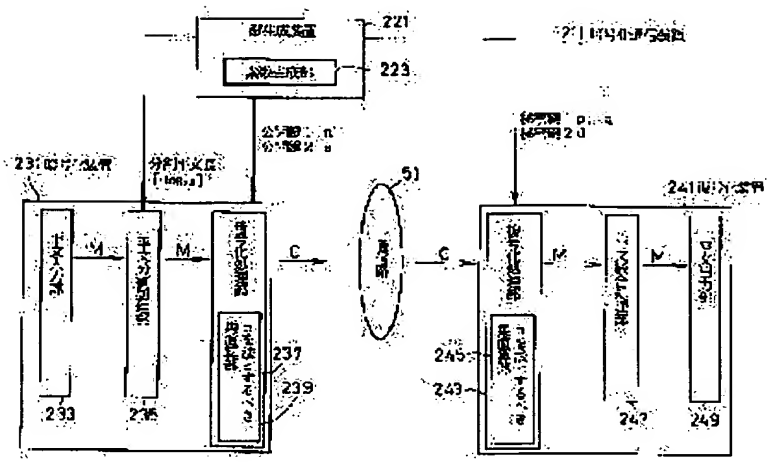
【図３１】



【図３２】



【図３３】



フロントページの続き

(51)Int.Cl.6		識別記号	庁内整理番号	F I		技術表示箇所
H04L	9/30			H04L	9/00	601F
	9/32					663B
						675B